Self-Service Data Preprocessing and Cohort Analysis for Medical Researchers

Jan Burmeister*
Fraunhofer IGD

Jürgen Bernard†
University of British Columbia

Thorsten May‡
Fraunhofer IGD

Jörn Kohlhammer §
Fraunhofer IGD
TU Darmstadt

Abstract

Medical researchers are increasingly interested in data-driven approaches to support informed decisions in many medical areas. They collect data about the patients they treat, often creating their own specialized data tables with more characteristics than what is defined in their clinical information system (CIS). Usually, these data tables or sEHR (small electronical health records) are rather small, maybe containing the data of only hundreds of patients. Medical researchers are struggling to find an easy way to first clean and transform these sEHR, and then create cohorts and perform confirmative or exploratory analysis. This paper introduces a methodology and identifies requirements for building systems for self-service data preprocessing and cohort analysis for medical researchers. We also describe a system based on this methodology and the requirements that shows the benefits of our approach. We further highlight these benefits with an example scenario from our projects with clinicians specialized on head&neck cancer treatment.

Index Terms: Human-centered computing—Visualization—Visual analytics——Social and professional topics [Health records]—

1 Introduction

Clinics all over the world collect data about the patients they treat in their clinical information systems (CIS). Physicians are increasingly using data-driven approaches to help understand and treat future cases. Oftentimes, clinical researchers specialized in certain diseases create own repositories in addition to their general CIS, that contain many more characteristics specific to those particular diseases. This is, firstly, due to inflexible data standards of their general CIS. Secondly, due to the fact that current research includes additional data usually not collected for a certain disease in the CIS by the clinic's procedures. Thirdly, certain clinics have specific standards for treatment decisions that are based on an extended set of data characteristics, especially for rare diseases. Finally, virtually every new clinical study poses new information, contexts, and data that are valuable to be recorded for downstream analysis. Hence, there are a vast number of locally collected, usually small, and very specific data tables including attributes about patients that are not defined by the clinic's standard procedures.

Medical researchers are working with many of such small electronic health records (sEHR) for cohort analysis. The challenge with such tabular data is not the size of the table, as the number of patients (especially for highly specific diseases) is rather small. In contrast, the main complexity of the data is the number, heterogeneity, multi-modality, and not least the quality of attributes (in the primary data). One major (and underinvestigated) problem in these projects, that we have observed in all of our design studies and in the VAHC's related work, is the time-consuming phase for data understanding and preprocessing as a basis for the later cohort analysis and result reporting. In fact, we claim that only a collaborative approach between medical researchers and data scientists can lead to a successful result. Anecdotally, one of our medical collaborators told us more than once that we simply do not have the medical knowledge to clean certain data attributes all by ourselves.

Accordingly, preprocessing is problematic due to several reasons. First, the data in these sEHR is collected by medical experts who use abbreviations and certain typology not only for attribute names but also for values of data fields. E.g., consider a boolean attribute for the positive result of a certain biomarker, which, however, is filled with entries such as ”considerable”, ”unobtrusive”, ”not sure”, ”irregular”, or ”re-check needed”. In addition to such uncertain statements, ambiguity is a major challenge such as, e.g., ”Y”, ”y”, ”YES”, ”J”, ”ja”, to indicate the value of a boolean flag. Such ambiguity problems become worse in collaborative settings with more than one expert. Therefore, a medical expert is needed who is involved in the creation of the sEHR to sort out certain questions. Second, not all of the data and information that is important to understand a disease and its progression is available in the sEHR (or any EHR for that matter). Medical expert knowledge is required not only to understand the data entries, but also during the process of creating reasonable cohorts, interpreting the analysis results, and distinguishing insights from well-known facts. Again, medical experts are needed as part of the preprocessing and analysis to also avoid a typical big data hubris that all questions can be answered with the right data analysis alone. Third, there is a lack of tools to support the medical environments we worked with, both for data preprocessing and cohort analysis. While tools like Tableau or Trifacta have had much success in the more IT-focused business analytics area, there are no similar tools that adhere to the constraints of sEHR and their typical clinical environment. All of the clinics we worked with have access to standard statistics tools like SPSS or SAS, while none of the medical researchers effectively use these tools.

Preprocessing is only one step towards successful medical analysis. Once the clinical data is transformed, medical researchers want to analyze their data, find interesting cohorts, compare these cohorts to each other, and communicate or even publish the result in their clinics or communities. However, these two steps for analysis and reporting are often not supported by the current IT environments of clinics. First and foremost, there is again a lack of tools to support the most common analysis functionality in a fashion that is accessible for medical researchers. In the clinics we have worked with, sEHR have typically been analyzed with general spreadsheet software or with the basic functionality of more powerful statistics tools. In both cases, the medical researchers could not perform the analysis by themselves but needed support from data scientists. The means for an autonomous cohort analysis would have considerable time benefits, at the very least.

From the reflection of our related work analysis and our previous projects, we infer that problems regarding data preprocessing and cohort analysis are omnipresent in many medical application areas. While the problems described above are present in many application areas, they seem exacerbated in medical use cases due to severe time constraints of highly specialized experts. The lack of tools that

*e-mail: jan.burmeister@igd.fraunhofer.de
†e-mail: jubernar@cs.ubc.ca
‡e-mail: thorsten.may@igd.fraunhofer.de
§e-mail: joern.kohlhammer@igd.fraunhofer.de

IEEE VAHC 2019
20 October, Vancouver, BC, Canada
978-1-7281-2019-5/19/$31.00 ©2019 IEEE

we encountered in many instances motivated the development of a methodology and a toolset with the goal to maximize the involvement of medical experts in data preprocessing and cohort analysis, while minimizing their exposition to IT-related details. With our methodology we aim to support the design of tools that cover the basic tasks of the complete medical data analysis pipeline, from data preprocessing over hypothesis testing to result reports. At the same time the medical researchers should be able to autonomously use the tool with as little data engineering knowledge as possible.

Our target groups are small medical research groups or individual experts with sEHR in the form of rather small, but complex, CSV or Excel files. At this stage, we support various forms of cohort analyses, with longitudinal analyses on the list for immediate future work. Our toolset covers the three phases of data preprocessing, stratification and hypothesis testing, and reporting. The preprocessing tool specifically targets data cleaning and transformation, which is typically done in a collaboration of medical experts and a data expert who knows the effects of certain preprocessing steps on the data side. The self-service tool for stratification, hypothesis testing and reporting is targeted at medical experts.

Our paper therefore aims at the following contributions:

- A new methodology that for the first time covers the entire collaborative analysis pipeline for medical experts preprocessing their data, analyzing cohorts and providing results.

- A proof of the viability of this methodology through two web-based prototypes, one for the preprocessing step and one for the hypothesis testing, cohort analysis and result presentation.

- A use case that is based on our collaboration with medical experts at the university clinic in Düsseldorf, Germany, and their team focusing on head&neck cancer, showing the benefits of the approach.

In the following, we provide further background on the related work, before we introduce the methodology. We will then introduce our system design, explaining both tools and how they interact. Our use case will then show the usage of our running system for medical experts, before we conclude and give an outlook on future work.

2 Related Work

We distinguish related work according to the different steps of our approach. Generic approaches have been presented and commercialized for all stages of the process. Solutions, such as, for example *Knime* (for explorative analysis) or *SPSS* (for statistical analysis) each offer a comprehensive toolkit to cover most, if not all scenarios along their analytical pipelines. Where possible, we also want to shed some light on solutions where degrees of freedom have been removed to allow for a less complex, more user-friendly approach to analysis.

2.1 Data Preprocessing

Weiskopf and Weng [19] present a survey on the data quality aspects most commonly addressed in medical literature. *Completeness* and *Correctness* are by far the most common issues that are addressed. Visual data preprocessing approaches can be distinguished by data type or by main goal (cleaning, restructuring, etc.). One of the earliest interactive approaches for tabular data is *Potter's Wheel* by Raman and Hellerstein [15]. Based on a spreadsheet, a user defines a series of data cleaning transformations either directly or by example. Deviations not matching the transformation rules are automatically identified and presented to the user for error checking. *Wrangler* by Kandel et al. [9] builds upon this idea. By interpreting user interaction on the spreadsheet, Wrangler suggests a set of potentially useful operators to be selected or modified by the user. The result of currently proposed operators is executed to be assessed by the user. *Profiler* by Kandel et al. [10] uses correlation analysis to identify causes for data anomalies. Several feature based detectors are used to identify potential problems. A set of linked views enables the user to locate and explore these anomalies in multivariate space.

Data preprocessing for time-oriented data became more prominent in the last decade. Bernard et al. [3] propose a editor for a linear transformation pipeline for time-series. A number of preprocessors can be selected, parametrized and inspected for all steps in the pipeline. While only few samples are used to build the process, this approach is reinforced against overfitting by automatically selecting the least similar samples to test and improve the pipeline. This concept has been extended recently [2] to multivariate time series. A user operates a modular, non-linear workflow which can be explored and controlled by visual sensors. They allow a visual input-output analysis connecting all stages of the workflow. Both approaches aim at preparing time-series for explorative analysis. In contrast, *Visplause* by Arbesser er al. [1] focuses on data-cleaning based on existing domain knowledge. Domain knowledge about sensor data, typical values and outliers and known dependencies are embedded an the metadata description. This representation drives the automatic identification of problems, which are summarized and explored with an interactive overview. Gschwandtner and Erhart [7] present an explorative approach to on data-cleaning. While this approach includes automated quality checks, a user may explore the views to identify patterns of quality problems.

2.2 Hypothesis Testing

de Prel et al [5] give an overview over the most commonly used statistical tests in six medical journals (1828 publications). Only three tests cover 70 percent of the publications. Thus, reducing the flexibility of a statistical toolkit in favor of a more accessible approach may in fact be a viable strategy. *Tourdino* [6] by Eckelt et al. is a recent approach to statistical hypothesis testing, deliberately reducing the number of options to be manually chosen. Tourdino operates on tabular data, and establishes a linearized approach to comparing rows or columns, which is mostly controlled by high-level decisions of the user. For example, suitable statistical tests and visualization types are derived from given comparison task and the data types of the relevant attributes. Their process supports non-experts in statistics with the most common scenarios. Subramanian's approach *VisiStat* [16] aims at performing and controlling statistical testing mostly by interacting with visualizations. As with TourDino, knowledge about suitable statistical tests is embedded in the mechanics of the approach. Wacharamanotham et al. [18] present and evaluate an approach which also guides a user through a partially interactive process. Their main goal, however, is to teach statistical testing instead of simplifying the process. In addition to test results, their system highlights the decisions made by the system, and the implicit assumptions that might lead to wrong conclusions.

2.3 Exploration

Exploration within EHR data often means the identification of related patient subsets, and the attributes that describe these subsets. In our experience, this task is still rarely supported by interactive tools. More often than not, exploration is driven by domain knowledge and performed through tedious, manual compilation of patient lists. *StratomeX* by Lex et al. [12] is a tool for the exploration and stratification of patients based on genomic data. The visualization combines multiple views each of which fitting a specific attribute type. It allows the interactive definition of stratification, their comparison, and the analysis of attribute dependencies. Bernard et al. [4] present a system that combines interactive cohort definition and correlation analysis. Any change to the cohort triggers an ensemble test with all attributes to point for potentially interesting dependencies. While this approach covers one temporal attribute, its design is tailored to a specific disease. *Outflow* by Wongsuphasawat and

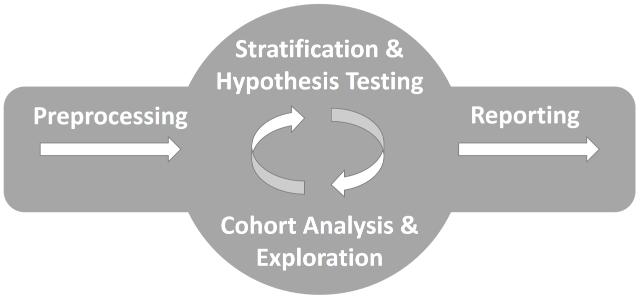


Figure 1: The four main components of our medical data analysis workflow

Gotz [20] helps identifying patient groups that share a common history. The dependency between patient events and outcomes is analysed in an aggregating view. Krause et al. [11] combine cohort specification and comparison with both temporal and non-temporal data. Remarkably, their primary goal is to provide training and test sets for building a predictive model, instead of doing a statistical test.

3 Methodology

Our experience across several years of research in various medical disciplines, and our review of currently available approaches (see above) has led us to use these valuable insights and design a methodology that combines the common workflow and requirements of these analysis projects. Based on this methodology we also implemented tools that support medical teams in preprocessing and analysis. Specific problems within the analysis pipeline have been addressed by the VAHC community, especially for specific diseases. It also shows that there is a need for a methodology that covers the entire analysis pipeline with a focus on the user group of medical researchers. The following methodology is designed to be used by future system approaches as a blueprint for building accessible tools by characterizing the users, the workflow and domain goals, and the typical external constraints that play an influential role on medical data analysis of sEHR.

3.1 Domain Characterization

3.1.1 Domain users

This methodology targets medical researchers, who are supported by data scientists during the data preparation phase. The group of medical researchers include senior medical experts as well as PhD students at a university clinic. The medical researchers want to focus on those tasks that require their medical knowledge to be performed. They are on a tight schedule, where research time competes with consultation and surgery hours. The collaboration ($\mathbf{R_4}$) between medical researchers and data scientists should therefore be as efficient (the least possible clarification time) and effective (cover all the required data questions) as possible. The medical researchers should be able to do as many of their meaningful tasks on their own as possible ($\mathbf{R_1}$). As an example, similar to physicians in previous projects, in our current head&neck cancer use case a PhD student wants to see correlations between different attributes in an sEHR to form new hypotheses, but does not want to browse and modify Excel tables to do this.

3.1.2 Workflow & Domain Goals

As depicted in Figure 1 we divide the workflow of medical data analysis into three natural stages: Data preprocessing, cohort analysis, and result reporting. The first stage of this pipeline is, given careful preprocessing, a one-time procedure, leading into stage two and three. Once the data basis is finalized, it is available for various studies, hence stages two and three are typically repetitive, open-ended processes. Sophisticated tools exist for each step of the pipeline, however they are typically focused to support data scientists with a technical background in mathematics and programming. Our toolset specifically supports medical researchers as further detailed below.

3.2 Requirements for sEHR Analysis Systems

The methodology covers a system environment for sEHR analysis that can be characterized by nine requirements:

- Medical researchers should be able to work with the sEHR analysis system autonomously most of the time ($\mathbf{R_1}$)
- For data preprocessing, the system should support various data cleansing and data transformation operations that can be performed by medical researchers in collaboration with data scientists ($\mathbf{R_2}$)
- A sEHR analysis system should support functionality for cohort creation, comparison and statistical analysis for confirmative analysis and various visualization and analysis techniques for exploration ($\mathbf{R_3}$)
- Such a system should support collaboration among medical researchers and between medical researchers and data scientists via storage and sharing features ($\mathbf{R_4}$)
- For a comprehensive sharing of results, the data cleansing and transformation operations, as well as the cohort analysis operations should be documented and stored with the created cohorts ($\mathbf{R_5}$)
- A sEHR analysis system should provide means to export and report the results as graphics familiar to medical researchers to be easily shared with colleagues in presentations or publications ($\mathbf{R_6}$)
- For such a system all operations on the data should be possible without the necessity for the data to leave the premise or network of the data owner ($\mathbf{R_7}$)

3.3 Data preprocessing

The researchers that approached us in the past always had specific questions in mind. Building on a basic knowledge about statistics and data visualization, we predominantly need to support the process to transform the raw data set into structured tables in the most straightforward way. The data preprocessing step involves the medical expert as a novice user, which is due to the fact that existing tools for data preprocessing are either unknown in the medical domain or excluding by requiring technical expertise with a background in programming. While existing tools allow very powerful modifications, we noticed that the most common issues in the medical domain can be solved by easily accessible interfaces.

The two primary tasks to be accomplished at this stage are data cleaning and data transformation:

Data cleaning Data cleaning, also referred to as data wrangling, builds upon general methodologies for data wrangling [8]. It highlights and resolves issues with the dataset on a table level, which are independent of the medical research question. The most common tasks we typically face with manually maintained data tables is fixing inconsistent or misspelled values, handling missing values, and removing attributes with problematic distributions or with content that does not contribute to any analysis. The result is a data base that fulfills mandatory integrity criteria for downstream transformation and analysis.

Data transformation Data transformation shapes and reduces the raw data into forms suitable for statistical analysis and visualization. In contrast to data cleaning tasks, data transformation is often guided by the actual medical research question. The data transformation task can be divided into attribute-based, i.e. more general, table-oriented operations, and value-based subtasks that address specific values:

- *Attribute-based*: Operations to change the name of an attribute or its data type. The names of attributes are oftentimes very cryptic and can only be resolved by medical researchers. At the same time, the data scientists can explain the pros and cons of certain data types, e.g. converting a text column with blood test results to a numeric data type. Frequently we also encounter categorical attributes with an implicit, natural order, e.g. the reported extent of smoking behavior, stages in cancer severity, or the intensity of therapies. Queries on such attributes using relational operators are quite natural, however this is not supported by common table-based file formats. Extended data storage formats are therefore required to properly encode this meta information for data processing algorithms.

- *Value-based*: Common tasks on the data entries themselves are merging overly specific categories to more general terms, or discretizing numeric attributes into categories. This also includes homogenizing values with different typing variants to a certain set of labels, and removing unit suffixes.

All of these operations on attributes and values should be recorded in an *edit history* that documents the sequence of operations (**R$_5$**). Building on lessons learned in data provenance, the history should be editable during data preprocessing, before it can be saved as a documentation for potential future use on similar data sets or on an extension of the same data set, a typical scenario in clinical research.

3.4 Cohort analysis

The entry point to medical analysis tools is often a specific question, usually concerning the correlation of attributes between two cohorts of patients (see Related Work). Assembling a cohort from a set of constraints sounds simple enough for a computer scientist, but turns into a cumbersome and time-intensive process for medical researchers with a general-purpose toolset. Providing an easy and fast way to define and compare cohorts is in our opinion the most crucial requirement and enabling factor for medical researchers. More specifically, from the related work and our previous projects, we take away two distinct user goals: Confirmative analysis for testing existing hypotheses, and exploration of cohorts for stratification and hypothesis identification.

Confirmative analysis In this case, medical researchers already have a specific question about their research data or a hypothesis. The main requirement here is to minimize the time needed to define a cohort, do visual and/or statistical correlation analysis, and to confirm or reject the hypothesis (**R$_3$**). The statistical methods should be familiar to medical researchers [5] and provide a visual interface that is easy to access. In fact, findings that were made with visualization tools were often the start for downstream analysis with other statistical tools that are e.g., used to calculate and report statistical significance. To provide such statistical functionalities as part of one tool is a major process simplification for medical researchers.

Exploration The second use case is oriented towards explorative research, where we see the long-term benefits of an established analysis platform (**R$_3$**). Once medical researchers become familiar with a medical analysis system - based on existing, or closed questions [14] - they start asking more open questions. Often, this was triggered by looking at a specific patient and expanding to the most similar patients from there, following a typical search paradigm as described by van Ham et al. [17]. Another important requirement is to support building new subcohorts from a currently active cohort, and to be able to find unexpected structures and distributions in the data. Finally, the ability to compare two or more cohorts and to easily assess the main differences between these cohorts is an often requested feature.

It is important to note that while these two goals are distinct, we believe there is a common base in which directed research leads to ideas for further exploration, which again triggers hypotheses that need to be confirmed or rejected (see Figure 1). An sEHR analysis system should therefore support both goals side by side.

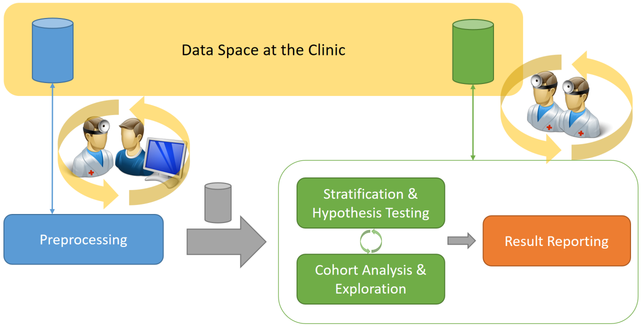

Data Space at the Clinic
Preprocessing
Stratification & Hypothesis Testing
Cohort Analysis & Exploration
Result Reporting

Figure 2: The medical data analysis workflow with our system

3.5 Result Reporting

The goal of most of the medical research work is to provide a result chart for presentation to colleagues and/or medical publications (**R$_6$**), maybe also as part of a dashboard in recurring meetings in the clinic. Any medical analysis tool sEHR should provide means for customizable charts that can be downloaded in a common picture format. There is the potential to link an sEHR tool to more sophisticated reporting tools maybe available on site, but a (reproducible) figure with the results of an analysis is sufficient in most cases.

3.6 External constraints

3.6.1 Documentation and Collaboration

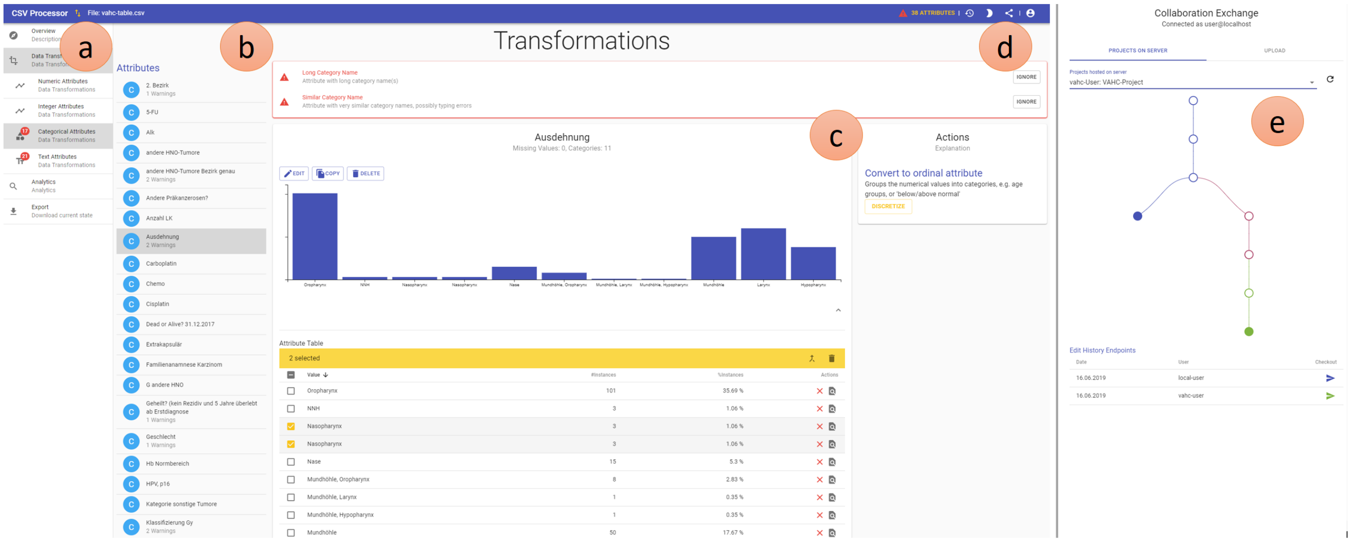
The ability to store cohorts and make cohorts reusable for future research and other colleagues is a key requirement for both documentation and collaboration (**R$_5$**) [9]. This goes in line with the requirement during preprocessing to document the changes to the data set before analysis. Both are indispensable for a later publication of a comprehensible analysis.

3.6.2 Data Governance

Medical data is subject to a high level of legal protection, especially if it could possibly be linked back to specific patients. The ethical approval process for granting permission to use such data for research projects varies slightly between medical institutions, and - in case of patient-related data - the consent that was given at the time the data was acquired. Even if the approval process is usually very well defined, we have learned this necessary ethical scrutiny to be a common source of delay. Therefore, the preferred way for all parties involved is to not transfer any data off premise at all, but to build distributed systems that can process the data on-site, i.e. within a secure IT network controlled by the data owner.

4 System Design

Based on our new methodology and the requirements identified in the previous section, we have designed and implemented a system for the analysis of sEHR. In the course of the system design, we have deliberately decided to split the three-stage pipeline into two separate



Long Category Name
Attribute with long category name(s)

Similar Category Name
Attribute with very similar category names, possibly typing errors

Convert to ordinal attribute
Groups the numerical values into categories, e.g. age groups, or 'below/above normal'

	Value	#Instances	%Instances	Actions
□	Oropharynx	101	35.69 %	✕
□	NNH	3	1.06 %	✕
☑	Nasopharynx	3	1.06 %	✕
☑	Nasopharynx	3	1.06 %	✕
□	Nase	15	5.3 %	✕
□	Mundhöhle, Oropharynx	8	2.83 %	✕
□	Mundhöhle, Larynx	1	0.35 %	✕
□	Mundhöhle, Hypopharynx	1	0.35 %	✕
□	Mundhöhle	50	17.67 %	✕

Date	User	Checkout
16.06.2019	local-user	➤
16.06.2019	vahc-user	➤

Figure 3: The preprocessing tool: The main sidebar (a) presents all attributes sorted by data type, the list (b) all corresponding attributes, with a note if an attribute is marked with warnings. For the selected attribute, the system has identified two issues (d) related to category names. By looking at the distribution (c), we note that two names (selected in the table), differ only in an invisible whitespace, and can thus be merged to one value. The distribution also reveals several categories with very few occurrences, actually special cases of the same term, as well as a mixture of German and Latin terms. We would propose to the medical researcher to review this particular attribute, perform the appropriate actions, and push the edits back to the online storage. The collaboration panel (e) shows a project that has been edited by three different users: The first user (blue) has initially started the project. User Red has decided to extend the process, leaving out Blue's last edit, and Green continued Red's progress.

applications. The preprocessing step has been moved into its own application, due to its nature as a one-time preliminary procedure and the specific collaboration between medical researcher and data scientist. In contrast, the cohort analysis is highly iterative and is targeted at the autonomous analysis work of medical researchers. Both applications provide collaboration features to store and exchange intermediate versions and results via web services (see Figure 2).

4.1 Preprocessing

The requirements gave us clear directions for the design of our system. Our main objective is to serve three primary goals: Guided data cleaning and objective-based data transformation (**R$_2$**) and collaborative means for medical researchers to work together (**R$_4$**).

4.1.1 Cleaning and Transformation

The data cleaning aspect is built around automatic warnings for common issues (**R$_2$**). After each modification, the system checks the data for issues, marks the respective attributes for user inspection, and provides solution suggestions. For this first iteration, we have focused to cover the most common issues we typically encounter, such as detecting overly long or possibly misspelled categorical values, uninformative distributions like single-category or free-text attributes, numeric outliers, and missing values. The guidance feature in its current form is primarily focused on explaining potential issues to medical experts using the system on their own and covers the most straightforward questions on data transformation (**R$_1$**).

Our system currently supports five data types (categorical, ordinal, float, integer, text), which are parsed automatically after a file has been loaded. We provide several tools to transform data, both for cleaning and objective-targeted purposes. The user starts with an overview page that provides statistical summaries and samples for each attribute. This serves as a point to detect high-level issues, such as renaming or deleting unnecessary or largely incomplete columns. Each attribute can be inspected in detail, with a visualization of its value distribution and data type-specific actions. A dedicated panel of warnings provides guidance, which of the possibly many attributes of the dataset actually needs inspection.

The most common actions to undertake is managing the values of categorical attributes, which are prone to typing errors and inconsistent terminology. Many analyses require numerical attributes to be normalized, outlier removed, or discretized into custom groups, as it is the case e.g. for age groups. Our system also provides specific support to create ordinal attributes from categorical and discretized numeric attributes, and define the underlying order. This data type is not automatically identifiable in a raw table data and requires explicit user initiative. From our experience though, we believe this additional effort pays of, since it allows improved queries and analysis options.

4.1.2 Export and Documentation

When finalized, the cleaned-up dataset can be submitted to the analysis platform but also saved again as a CSV file. In this way, the tool can also be used in a stand-alone manner, which is useful if users want to use other analysis software for the next step. Also, the history of edits actions can be saved, as documentation how exactly the original dataset has been modified (**R$_5$**).

4.1.3 Collaboration

Our ultimate goal is to enable the medical researchers to complete this process completely on their own. On the other hand, in most cases it is advantageous to execute the data preprocessing in a collaborative manner with a data scientist (**R$_4$**). To facilitate this step of the process, we provide a web service for online collaboration. At any point in the process, a user can upload the current state of the work to an installation of this web service, ideally installed on a secured server controlled by the data owner (**R$_7$**). Such a project on the server consists of the original dataset and the tree of all edit actions that the users have performed and committed. Authorized users can access the working tree and download the project at any point in the edit history. The application will download the original data and apply the sequence of edit actions to reproduce the requested state.

The user can then continue at this state, and upload own changes to the server, thereby adding to the tree of edits.

4.2 Analysis Platform

While the preprocessing platform is primarily intended to create intermediate versions and the final version of the data set for the following analysis, the analysis platform provides open-ended analysis services to many individuals working independently on the same dataset (**R$_3$**). Each dataset is by design immutable to allow the compatible storage of cohort definitions to reuse and share results with other researchers.

The platform's core design concept is built around creating, sharing and re-using cohorts. The VA community has presented numerous ways to define cohorts and to analyze them. Following a platform-oriented architecture, these techniques could be additional or alternative building blocks of a future system based on our methodology. In this first prototype, we provide two modules to define a cohort, and several views to inspect them. At any time, users can save the created cohorts and compare their findings with the database of cohorts that have already been created in the current or previous working sessions or by colleagues.

4.2.1 Stratification & Hypothesis Testing

This first approach is a direct implementation of an interface for medical researchers who want to answer a specific question. Our interface consists of three parts: A panel to define two cohorts via a list of constraints, a panel that visualizes selected attribute distributions and correlations of the created cohorts, and a panel that embeds the two cohorts into the database of already defined cohorts. Figure 4(a), for example, shows two defined cohorts on the left. The client web application maintains a steady websocket connection to the worker server, where the actual processing is done. All components of the interface are individually connected to resources on the server in an asynchronous manner. Whenever the user modifies e.g. the cohort definition, the affected server resources are notified, and the interface components update as soon as the server can push the new results. This way we can provide quick feedback, e.g. showing how a new constraint affects the distribution a specific cohort attribute, while more extensive components might still be in the computing stage.

Constraint-based cohorts The cohort constraints are resembling the filter conditions that the medical researcher previously had to evaluate manually in a spreadsheet software. These include value selections for categorical attributes, and ranges, thresholds or exact matches for numeric attributes. A notable addition is the support for the special nature of ordinal, or sorted categorical attributes, which is common in data within the medical domain. The overview cards (see Figure 4(a)) present a concise list of these constraints, each with a short description and a colored bar below the description. The length of the bar indicates the size of the cohort (percentage of the subset compared to the entire dataset) under these constraints.

Cohort analysis The four chart views (see Figure 4(b)) each depict the distribution of a selected attribute, either for only one of the two cohorts or both side-by-side. Alternatively, two attributes can be plotted against each other to check for possible correlations. A medical researcher with a specific question in mind can use this page to quickly define a cohort, assess its properties, and compare them against the overall population or the second user-defined cohort. The two cohort definitions can be named and stored on the server - for this session only or permanently in a database. All definitions can be copied back into one of the two cohort slots.

The third panel of this interface shows all of these server-stored cohorts (Figure 4(c)). This panel serves as a cohort browser, but also displays the automatic part of the cohort analysis. After every cohort modification, the server computes the intersections between the two cohorts with all those on the server, and highlights those above a certain similarity threshold. The top, green line under the cohort name shows the similarity of this cohort to the first cohort, the bottom, blue line the similarity to the second cohort. In this way, the system might identify unexpected similarities between the current cohorts and another one, probably defined in a quite different way at an earlier time. Also, the server computes the distributions for all attributes of the two cohorts, and displays them in a ranked list, either alphabetically or by similarity. We have placed these features in a sidebar, in order to not distract the user from the current analysis goal. It is always available at a side-glance to provide additional cues to direct the user in further unexplored directions.

4.2.2 Cohort Exploration

This approach is centered around building cohorts while being in an exploration-oriented workflow (not depicted in Figure 4).

Similarity-based cohorts In a similarity-based approach, users start with a set of attributes and query values, e.g. characterizing a specific or prototypical patient case. The system then searches for the most similar patients, and presents the overall distribution of each query attribute, with markers depicting the query value and those of the result set. The user can choose to add an additional range filter on each attribute, to e.g. exclude outliers that are too far away from the query value, thereby refining the cohort of patients based on value similarity. This approach supports especially clinicians who wish to find similar cases for a current patient as an additional, data-driven aid to improve decisions.

Projection-based cohorts Another, more visual approach, is driven by data projection methods, in our case t-SNE [13]. Given a set of n user-selected attributes, the technique places the n-dimensional data points of a patient population on a 2D plane. As usual for dimension reduction, information is lost in this process. Nevertheless, the technique can provide useful insights, revealing clusters or other structural properties within the data. The user can lasso-select a group of points in the projection (e.g. an outlier cluster), and thereby create a cohort based on structural similarity.

Both similarity-based approaches are accompanied with views to inspect attribute distributions of the selected cohort, again also in relation to the existing set of stored cohorts. The similarity cohorts can be stored and also loaded into the comparison view as another type of constraint.

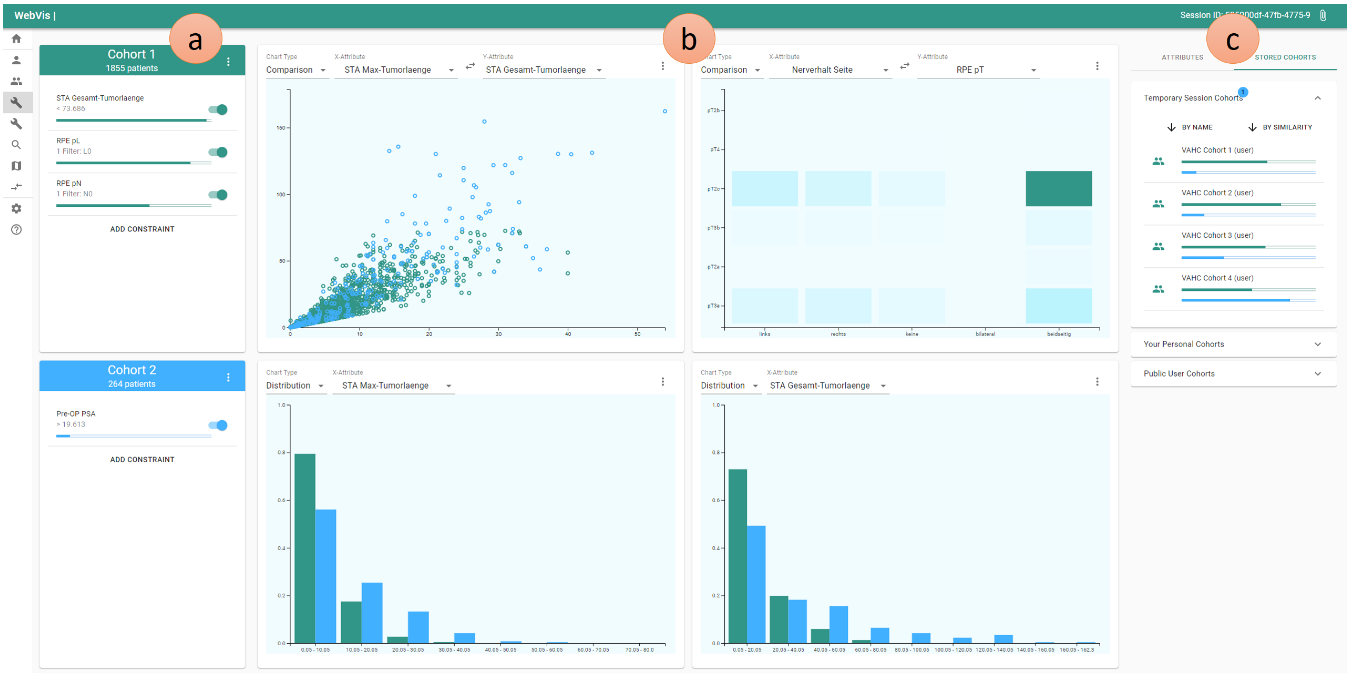
4.2.3 Result Reporting

To complete the data analysis process, all charts can be customized and downloaded to the user's local drive. Currently we provide options for size, color, fonts and background grids, so that the charts can be individualized to e.g. satisfy a journal's specification, or match charts created by another application (**R$_4$**).

4.3 Distributed Data Management

The system architecture has been designed to ensure that all data is stored and processed exclusively within the boundaries of the owner (**R$_7$**). Our system is built around *domain nodes*, a self-contained bundle of web services for data storage, access control and processing. This bundle can e.g. be installed by an institute's department to provide analysis services for all associated researchers and students. To simplify user management, all domain nodes register themselves at a central registry service, which provides a global user authentication service. In this way, users only need a single account to authenticate to all preprocessing or analysis servers across e.g. a university faculty, but the access rights to the actual data is controlled by the individual node maintainers, e.g. the medical departments.

Cohort Exchange We have only shown a few ways how to define cohorts in this work, but there are many more. Some, such as a list of attribute constraints, can be re-applied to e.g. a modified cohort with added patients. Others cannot be re-applied, such as the



Cohort 1
1855 patients

STA Gesamt-Tumorlaenge
< 73.686

RPE pL
1 Filter: L0

RPE pN
1 Filter: N0

Cohort 2
264 patients

Pre-OP PSA
> 19.613

Chart Type: Comparison
X-Attribute: STA Max-Tumorlaenge
Y-Attribute: STA Gesamt-Tumorlaenge

Chart Type: Comparison
X-Attribute: Nerverhalt Seite
Y-Attribute: RPE pT

Chart Type: Distribution
X-Attribute: STA Max-Tumorlaenge

Chart Type: Distribution
X-Attribute: STA Gesamt-Tumorlaenge

Temporary Session Cohorts
VAHC Cohort 1 (user)
VAHC Cohort 2 (user)
VAHC Cohort 3 (user)
VAHC Cohort 4 (user)
Your Personal Cohorts
Public User Cohorts

Figure 4: The primary page to define and compare cohorts. The two cards on the left (a) provide the tools to define the list of constraints for two cohorts (the actual constraint configurations are available in a popup). The views in the center (b) are used to assess selected attribute properties, and the panel on the right (c) contains all controls for the server-related content, i.e. the cohort browser and computed similarities.

projection-based cohort, which is built by a specific user selection on a 2D canvas. The minimal common baseline to store and exchange cohort definitions is by a set of IDs, indicating which patients are included in the cohort. This restricts datasets to be immutable, to ensure such an ID-based cohort definition stays compatible at all times. We decided to accept this as a trade-off, because this generic approach enables us to include basically any cohort definition technique. This, by design, also allows simple and fast queries in data processing algorithms. We store the IDs as a binary sequence, along with optional meta data for application-specific information on how the cohort was built. For example, the constraint-based cohorts described above store the actual constraints, allowing the user to reload these cohorts with their exact definition in the client. Another benefit of this approach is that we can open the database service to external applications (**R$_4$**). The interface that our platform services use internally to access the hosted datasets and cohort definitions can be used in the same manner by external systems, thereby enabling the exchange of cohort definitions across application boundaries.

5 Example Scenario

This section describes a scenario to highlight the benefit of the methodology and the usage of our system. For this, we will start with an example analysis that was done with our collaborators at the university clinic in Düsseldorf before the new system was in place. The second scenario shows how we are able to collaboratively work with the system.

In this particular case, our partner gave us a pseudonomized (for us, anonymized) data table with 300 patients who have or had head&neck cancer. The table had 159 attributes with a mix of numerical and text formats. The attribute names were sometimes abbreviations that only made sense to medical researchers. The textual values contained a mixture of German and Latin names of the diseased regions in the head&neck area. It took us several weeks of online meetings to agree on one cleaned version of the dataset that we could use as foundation to start building the actual data analysis system. Lacking domain knowledge, we were extremely hesitant to modify any information or delete rows with missing values. Since clinicians have a dense schedule, it understandably took a while for any of our questions to be answered, so that we could proceed with the data transformation. We finally ended up with a large number of non-transferable rules coded in a programming language, defining which column values should be merged to more general terms or actually replaced as missing values, and which attributes should be renamed or removed in subsequent versions of the data.

We actually came back to this dataset for this paper, this time executing the pipeline with our system. Of course we were familiar with this dataset, but the execution time to set up rules and constraints was significantly lower and better documented. Starting with the preprocessing tool, we quickly removed a large number of irrelevant data columns, and renamed all others to more meaningful, readable titles. In one particular column that was discussed quite a bit back then, we merged several overly specific medical classifications - that occurred only very rarely, messed up our visualizations, and were therefore often excluded - to their broader category. The whole process was done without any programming. Afterwards, we could directly use the data on our analysis platform, evaluating if our ideas for this project would work on the given data basis.

Figure 3 shows the transformations during the data preprocessing. We started out inspecting the overview table of all attributes, and removed some rather pointless attributes, such as an index column, or a free-text column for notes. The automatic warnings then directed us to several categorical attributes. When selecting the attribute ”Extent” (German: Ausdehnung), the system pointed us to some single-character typos in complex terms, including a case that differed only in a trailing whitespace (see the two entries on *Nasopharynx*, highlighted in Figure 3) - a difference that is literally invisible in a spreadsheet software, but would still turn out to be different categories in string-processing algorithms. Based on word distance metrics, our system issued respective warnings, and the problem could be solved by merging the categories together with a few mouse clicks. Looking at the distribution chart, we also noted that there were several categories with barely any occurrences. Upon inspection, all of those were special cases of a more general term, also present itself with a much higher occurrence. We asked the medical researcher to evaluate this particular attribute, and probably combine these specialized terms into one. We concluded the preprocessing by binning a column of patient ages into a set of common groups, and removed the original column.

We are convinced that in the future these online collaboration and documentation features will help us and other researchers following our methodology to deal with data preprocessing and analysis processes much faster and in a more transparent way for all parties involved. We believe that including the medical experts right from the start of a data science project, enabling them to follow and take part in the process via an easy to use interface, is an immensely valuable foundation for the whole lifetime of a joint research project. Our collaboration partner was excited about the prospect of autonomous cohort analyses for themselves and their team.

6 Conclusions and future work

In this paper, we have introduced a new methodology for building cohort analysis systems that allow a self-service data preprocessing and cohort analysis of sEHR for medical researchers. Based on an analysis of the related work, we have identified nine requirements for such systems and have further detailed the steps of the entire pipeline from raw data to result reporting. A section on the system design and an example scenario from an actual project with clinicians show the benefit of both the methodology and a respective system based on this methodology.

In our future work, we will especially work on extending the general system capabilities. The visual analysis components that were presented in the system section were all built and tested in previous projects, and there are several more components that have not been linked in one single platform. We currently work on extending this prototypical version, both in analysis functionality and data type support. We are also especially keen to extend the capabilities of the preprocessing tool, which serves as a gateway to our platform. On the one hand, we want to extend the capabilities of the automatic warnings, to catch more and also less frequent issues. On the other hand, we want to handle more advanced data transformation tasks that are usually performed by writing code in script languages. Notable examples are the creation of derived attributes, combining the data of multiple columns into one result. We are considering approaching this via concepts from graphical programming languages, thereby describing the transformations similar to flow diagrams that are well-known in the medical domain. Another difficult problem is resolving implicit encodings. We encountered several examples where entries in the data tables were missing - not because the value was actually unknown, but because the clinicians did not enter them, as they could infer the condition from other attributes. Resolving such cases requires complex logic on multiple attributes, which is an open challenge to solve in a graphical user interface. We also aim to provide improved methods to replicate the preprocessing steps on an extended or modified dataset in a quick and safe way.

Acknowledgments

This project has received funding from the European Unions Horizon 2020 research and innovation programme under grant agreement No 731944. The content of this publication reflects only the views of its authors. The European Commission is not responsible for any use that may be made of the information it contains.

References

[1] C. Arbesser, F. Spechtenhauser, T. Mühlbacher, and H. Piringer. Visplause: Visual data quality assessment of many time series using plausibility checks. *IEEE TVCG*, 23(1):641–650, Jan 2017. doi: 10.1109/TVCG.2016.2598592

[2] J. Bernard, M. Hutter, H. Reinemuth, H. Pfeifer, C. Bors, and J. Kohlhammer. Visual-Interactive Preprocessing of Multivariate Time Series Data. *Computer Graphics Forum*, 2019. doi: 10.1111/cgf.13698

[3] J. Bernard, T. Ruppert, O. Goroll, T. May, and J. Kohlhammer. Visual-interactive preprocessing of time series data. In *Proc. of SIGRAD 2012; Interactive Visual Analysis of Data; Växjö; Sweden*, vol. 81, pp. 39–48. Linköping University Electronic Press; Linköpings universitet, 2012.

[4] J. Bernard, D. Sessler, T. May, T. Schlomm, D. Pehrke, and J. Kohlhammer. A visual-interactive system for prostate cancer cohort analysis. *IEEE Computer Graphics and Applications*, 35(3):44–55, 2015. doi: 10.1109/MCG.2015.49

[5] J.-B. du Prel, B. Roehrig, G. Hommel, and M. Blettner. Methods and dimensions of electronic health record data quality assessment: enabling reuse for clinical research. *Dtsch Arztebl Int 2010*, 107(19):343–350, 2010. doi: 10.3238/arztebl.2010.0343

[6] K. Eckelt, P. Adelberger, T. Zichner, A. Wernitznig, and M. Streit. Tourdino: A support view for confirming patterns in tabular data. *EuroVis Workshop on Visual Analytics (EuroVA '19)*, 2019.

[7] T. Gschwandtner and O. Erhart. Know your enemy: Identifying quality problems of time series data. In *IEEE PacificVis 2018, Kobe, Japan*, pp. 205–214, 2018. doi: 10.1109/PacificVis.2018.00034

[8] S. Kandel, J. Heer, C. Plaisant, J. Kennedy, F. van Ham, N. H. Riche, C. Weaver, B. Lee, D. Brodbeck, and P. Buono. Research directions in data wrangling: Visualizations and transformations for usable and credible data. *Information Visualization*, 10(4):271–288, Oct. 2011. doi: 10.1177/1473871611415994

[9] S. Kandel, A. Paepcke, J. Hellerstein, and J. Heer. Wrangler: Interactive visual specification of data transformation scripts. In *CHI 2011*, pp. 3363–3372. ACM, New York, NY, USA, 2011. doi: 10.1145/1978942.1979444

[10] S. Kandel, R. Parikh, A. Paepcke, J. Hellerstein, and J. Heer. Profiler: Integrated statistical analysis and visualization for data quality assessment. In *Advanced Visual Interfaces*, 2012.

[11] J. Krause, A. Perer, and H. Stavropoulos. Supporting iterative cohort construction with visual temporal queries. *IEEE TVCG*, 22(1):91–100, Jan 2016. doi: 10.1109/TVCG.2015.2467622

[12] A. Lex, M. Streit, H.-J. Schulz, C. Partl, D. Schmalstieg, P. Park, and N. Gehlenborg. Stratomex: Visual analysis of large-scale heterogeneous genomics data for cancer subtype characterization. *Computer Graphics Forum*, 31(3):1175–1184, 2012. doi: 10.1111/j.1467-8659.2012.03110.x

[13] L. v. d. Maaten and G. Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[14] G. Marchionini. Exploratory search: From finding to understanding. *Commun. ACM*, 49(4):41–46, Apr. 2006. doi: 10.1145/1121949.1121979

[15] V. Raman and J. M. Hellerstein. Potter's wheel: An interactive data cleaning system. In *Proc. of the 27th Intl. Conf. on Very Large Data Bases*, VLDB '01, pp. 381–390. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2001.

[16] K. Subramanian. Visistat: visualization-driven, interactive statistical analysis. In *CHI 2014: Extended Abstracts on Human Factors in Computing Systems*, pp. 987–992, April 2014. doi: 10.1145/2559206.2579423

[17] F. van Ham and A. Perer. Search, show context, expand on demand: Supporting large graph exploration with degree-of-interest. *IEEE TVCG*, 15(6):953–960, Nov 2009. doi: 10.1109/TVCG.2009.108

[18] C. Wacharamanotham, K. Subramanian, S. T. Völkel, and J. Borchers. Statsplorer: Guiding novices in statistical analysis. In *CHI '15*, pp. 2693–2702. New York, USA, 2015. doi: 10.1145/2702123.2702347

[19] N. G. Weiskopf and C. Weng. Methods and dimensions of electronic health record data quality assessment: enabling reuse for clinical research. *Journal of the American Medical Informatics Association : JAMIA*, 20(1):144–151, Jan-Feb 2013.

[20] K. Wongsuphasawat and D. Gotz. Exploring flow, factors, and outcomes of temporal event sequences with the outflow visualization. *IEEE TVCG*, 18(12):2659–2668, Dec 2012. doi: 10.1109/TVCG.2012.225