UNIVERSITY OF TARTU
Faculty of Science and Technology
Institute of Technology

Uscinnia Dyn’ko

Medi_Vizz: A tool for concurrent visualisation of medical data

Bachelor’s Thesis (12 ECTS)

Curriculum Science and Technology

Supervisor(s):
Associate Professor of Genomics and Metabolomics Toomas Haller
MSc Miriam Nurm

Tartu 2024

Medi_Vizz: A tool for concurrent visualisation of medical data

Abstract:

Doctors must analyse the extensive amount of medical data before diagnosing a patient and prescribing treatment. This is especially complicated for patients with multiple chronic conditions, who can be treated by different specialists. The advent of electronic health records granted doctors the opportunity to have all the available information at hand – on the computers at their workstations. However, the challenge of representing these data in an easy-to-understand way remained. The Medi_Vizz application allows doctors to visualise individual patients' diagnoses and laboratory analysis data. Medi_Vizz has a user-friendly interface and creates comprehensible plots with medical data. Doctors can utilise these plots in clinical decision support, while scientists can examine disease progression in individual patients, making Medi_Vizz a valuable tool for both general healthcare and scientific research.

Keywords:

Electronic health records (EHRs), Multimorbidity, Disease trajectory, Medical data visualisation, Medical informatics.

CERCS:

B110 Bioinformatics, medical informatics, biomathematics, biometrics;

Medi_Vizz: Tööriist meditsiiniliste andmete samaaegseks visualiseerimiseks

Lühikokkuvõte:

Arstide jaoks on patsiendi diagnoosimiseks ja ravi määramiseks väga oluline analüüsida ulatuslikke meditsiinilisi andmeid. See on eriti keeruline mitme kroonilise haigusega patsientide puhul, keda ravivad erinevad spetsialistid. Elektrooniliste terviseandmete (EHR) kasutuselevõtt on andnud arstidele võimaluse koondada kogu vajalikku teavet ühes kohas - oma töökoha arvutites. Kuid nende andmete hõlpsasti mõistetaval viisil visualiseerimise väljakutse on jätkuvalt aktuaalne . Käesolevas töös välja töötatud rakendus Medi_Vizz võimaldab arstidel visualiseerida individuaalsete patsientide diagnoose ja laborianalüüside andmeid ajateljel. Medi_Vizz-il on kasutajasõbralik liides ning see loob arusaadavaid graafikuid meditsiiniliste andmete põhjal. Arstid saavad neid graafikuid kasutada kliinilise otsuste tegemiseks, samas kui teadlased saavad uurida haiguste kulgu individuaalsete patsientide tasemel, muutes Medi_Vizz-i väärtuslikuks tööriistaks nii tervishoius kui ka teadusuuringutes.

Märksõnad:

Elektroonilised terviseandmed (EHR), Multimorbiidsus, Haiguse trajektoor, Meditsiiniliste andmete visualiseerimine, Meditsiiniline informaatika.

CERCS:

B110 Bioinformaatika, meditsiiniinformaatika, biomatemaatika, biomeetrika;

TABLE OF CONTENTS

ABBREVIATIONS

EHRs – Electronic Health Records

CDS – Clinical Decision Support

ICD – International Statistical Classification of Diseases and Related Health Problems

NLP – Natural Language Processing

EstBB – Estonian Biobank

NPR – The Danish National Patient Registry

RR – Relative Risk

OMOP CDM – Observational Medical Outcomes Partnership Common Data Model

OHDSI – Observational Health Data Sciences and Informatics

INTRODUCTION

One of the fundamental parts of the doctor's profession is to assess the extensive amount of information available for an individual patient to make a diagnosis. With paper-based medical records, it was a complicated task to get the whole picture of the patient's anamnesis, especially for patients with multiple chronic conditions. To ensure accurate diagnosis and appropriate treatment prescription, it is important to consider how one condition can affect the other and how different treatments may interact. The standard approach to treating such patients is to focus on the most serious condition and treat one disease at a time, leading to overlooking milder conditions. The arrival of Electronic Health Records (EHRs) allowed doctors to access all of the available data of a patient from their workspace computer, providing a holistic view of the patient's medical history. Even though data became easy to access, the question of representing these data in a clear format arose. The level of abstraction should allow the inclusion of several data formats in one picture, be straightforward to comprehend, and aid in diagnosing patients and prescribing treatment.

This thesis focuses on developing a tool (Medi_Vizz) for concurrent medical data visualisation. Medi_Vizz assists doctors in viewing and analysing the medical history of individual patients. For scientists, this tool supports the study of the progression of diseases and their symptoms. Medi_Vizz combines diagnosis and laboratory measurement data into a concise and intuitive visual representation. This application addresses the issue of summarising the growing amount of medical data for utilisation in general healthcare and enhancing research on disease development in individuals.

1 LITERATURE REVIEW

The digitalisation of medical records in the 20th century brought EHRs into being for doctors to help diagnose their patients and for scientists to broaden their research objectives. Doctors were allowed to have all the information connected with their patients in one place and utilise this for clinical decision support (CDS). This created a need for software that would visualise EHRs and aid doctors in diagnosing patients by showcasing several layers of information in one representation. On the other hand, for scientists, this signalled the potential extraction of helpful disease development patterns from the vast amount of unprocessed data to further study the diseases' pathology and causes.

In this chapter, EHRs are described as a source of data for doctors (Section 2.1), and two viewpoints are introduced for utilising EHRs: scientific (Section 2.2) and doctors' (Section 2.3), with the description of the software developed for both perspectives.

1.1 Electronic health records

The medical record is a term used to describe written medical information assigned to an individual patient during routine healthcare delivery [1]. The first medical records can be traced back to several thousands of years BC. Nevertheless, consistent medical records use was not known until the beginning of the 20th century. For example, clinicians at the Mayo Clinic started assigning patient numbers and recording patients' information in 1907 [2]. Traditionally, medical records were one-copy, paper-based documents stored systemically in hospital archives. With the development and accessibility of technology and computers, the gradual switch to digitalised medical records – EHRs began.

Initially, doctors and nurses struggled with data entry [3, 4], and by the end of the 20th century, a combination of paper-based and electronic medical records was used. Such initiatives as the Health Information Technology for Economic and Clinical Health Act in the USA in 2009 [5] or the Estonian Nationwide Health Information System in 2008 [6] encouraged healthcare providers to implement EHRs. EHRs allowed to store vast amounts of patient information, eased doctors' access to medical data, and improved coordination between healthcare facilities. However, they also revealed how unsuited the information in medical records was for research studies. First, the need to standardise the data became apparent; second, secure data exchange and consent issues arose [7].

According to a review by Jensen *et al*., EHRs include various types of data, from drug prescriptions with dosages to a doctor's notes on the reasoning behind the prescription, the

patient's lifestyle, and well-being. Additionally, three clinical data types are captured in EHRs [8]:

- ***Administrative data*** serve administrative purposes and aid in supervising hospitals' work and reimbursing government and insurance. Such data include age, sex, ethnicity, address, and classified diagnoses. Since these data are omnipresent, they become a preferable data source for population-based health research.
- ***Ancillary clinical data*** are provided by laboratories, pharmacies, radiological or medical imaging departments, and sequencing centres.
- ***Clinical text*** documents the decision-making behind the diagnosis and prescriptions and summarises patients' treatment plans by consultation with medical protocols and clinical documentation. These are the most challenging data types to analyse due to text variations, spelling and grammar errors, and non-standard abbreviations.

1.1.1 ICD-10 classification

International Classification of Disease translates diagnosis codes into alphanumeric code, creating categorised medical data that are easy to understand in different settings, convenient for storage and retrieval and usable in research. Attempts to implement the classification of diseases began centuries ago, with some researchers tracking the first one to the 18th century. The system has now evolved into the international standard for diagnosis codes, the International Statistical Classification of Diseases and Related Health Problems (ICD). Initially, classification was developed for recording the cause of death in documents, and later, it expanded to include morbidity causes – diseases. Modern ICD is intended not only to categorise diseases and injuries but also to encode any admission and interaction in the healthcare facility. Examples are symptoms, signs of the disease, abnormalities, reasons for doctor consultation, reasons for admission, and diagnosis [9].

ICD is revised and updated every decade by the World Health Organisation [10]. More than 100 countries use the ICD-10 system for mortality classification, including Estonia, which adopted the 10th version of the ICD in 1997 [11]. The ICD-10 system is much more specific than the previous version: the system expanded from 17,000 codes in the ICD-9 to nearly 155000 in the ICD-10. The 9th version used 5-digit codes to specify the disease, while the 10th version uses seven digits. For example, 255 cases were identified where one ICD-9 code was mapped to 50 ICD-10 codes [12]. ICD-10 classification is divided into 22 chapters, and the first letter of the code encodes the chapter (Table 1) [9]. Three-character categories constitute each chapter to cover all the diagnoses, each subsequent character introducing greater specificity (Fig. 1) [12].

Table 1. Chapters of the ICD-10 classification with the respective codes and diagnoses [9].

ICD-10 Chapter	Codes	Diagnoses
I	A00-B99	Certain infectious and parasitic diseases
II	C00-D48	Neoplasms
III	D50-D89	Diseases of the blood and blood-forming organs and certain disorders involving the immune mechanism
IV	E00-E90	Endocrine, nutritional and metabolic diseases
V	F00-F99	Mental and behavioural disorders
VI	G00-G99	Diseases of the nervous system
VII	H00-H59	Diseases of the eye and adnexa
VIII	H60-H95	Diseases of the ear and mastoid process
IX	I00-I99	Diseases of the circulatory system
X	J00-J99	Diseases of the respiratory system
XI	K00-K93	Diseases of the digestive system
XII	L00-L99	Diseases of the skin and subcutaneous tissue
XIII	M00-M99	Diseases of the musculoskeletal system and connective tissue
XIV	N00-N99	Diseases of the genitourinary system
XV	O00-O99	Pregnancy, childbirth and the puerperium
XVI	P00-P96	Certain conditions originating in the perinatal period
XVII	Q00-Q99	Congenital malformations, deformations and chromosomal abnormalities
XVIII	R00-R99	Symptoms, signs and abnormal clinical and laboratory findings, not elsewhere classified
XIX	S00-T98	Injury, poisoning and certain other consequences of external causes
XX	V01-Y98	External causes of morbidity and mortality
XXI	Z00-Z99	Factors influencing health status and contact with health services
XXII	U00-U85	Codes for special purposes

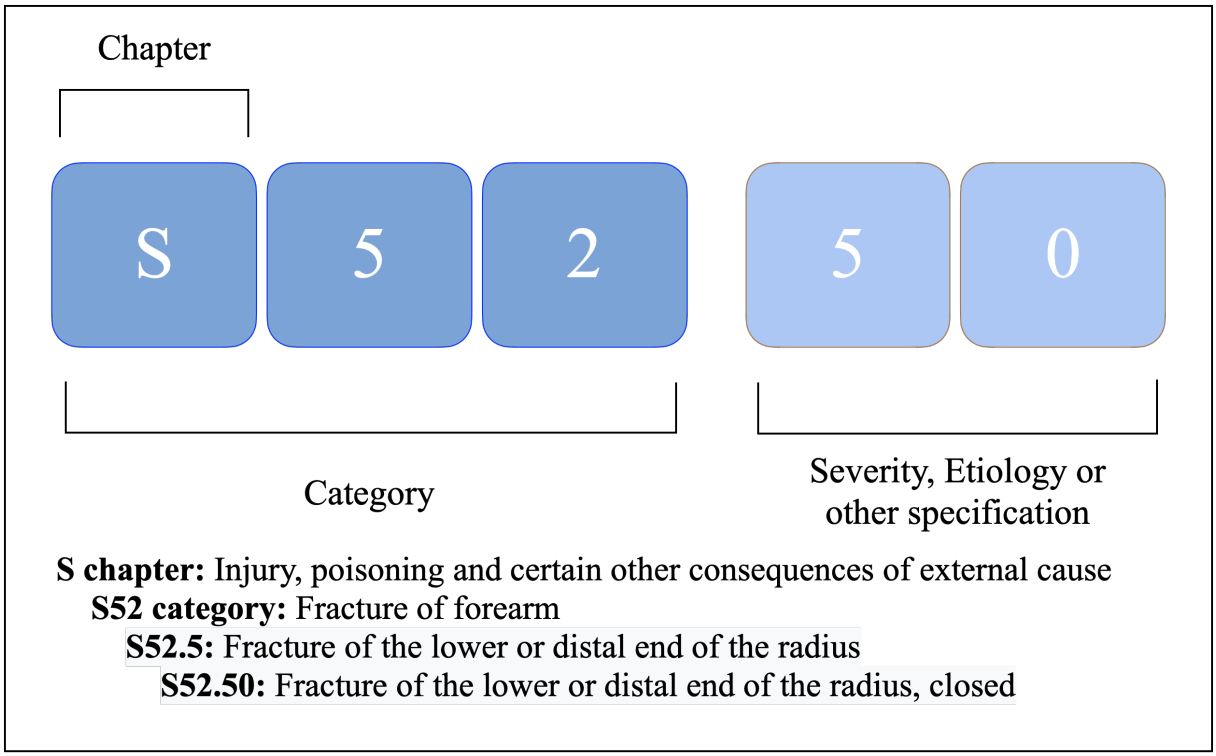

Fig. 1. Structure of an ICD-10 diagnosis code based on the forearm fracture. Figure adapted from Deschepper *et al*. [13] and modified with the ICD-10 code used in Estonia [14].

1.1.2 Laboratory measurements

In many cases, CDS is impossible without laboratory medicine and diagnostics. Laboratory medicine examines the composition and concertation of analytes in biological samples (fluids and other) [15]. Examples of analytes can be biochemical compounds (electrolytes, metabolites, tumour markers, hormones, other proteins), immunological markers (antibodies), microorganisms (viruses and bacteria), laboratory haematology parameters (blood cell count, body fluid cell count), genetic markers (sequencing, karyotyping). The analytes can be evaluated in venous, capillary, or arterial blood, urine, cerebrospinal fluid, tissue samples and other body fluids [16,17].

After the patient's sample analysis has been performed, the result is compared to the reference range. The reference range is a normal range for the analyte in the sample, which the doctor uses to compare with the patient's measurement. The reference range is based on the values met in 95% of the healthy population [18]. It is important to note that reference ranges depend on sex, age, when the test was performed, and the type of biological material used [17].

At the beginning of the 20th century, it was a common belief that ~70% of clinical diagnoses were based on laboratory data and constitute the same percentage of EHRs. This fact became known as the "70% claim". Later, this claim was refuted due to a lack of evidence, but the

importance of laboratory medicine cannot be understated [19]. Automated laboratory analysers can carry out multiple analyses quickly and at a low cost, but they provide fundamental data for CDS, treatment, management and prevention of the disease [20].

1.1.3 Analysis of the EHRs

EHRs can be structured (administrative data, laboratory results, medications) or unstructured (medical images and clinical text). Structured data analysis is more straightforward, and statistics and machine learning methods can be applied to it. On the other hand, unstructured data require a lot of preprocessing due to the difficulty of extracting information from sources that include a lot of context and uncertainties in reporting.

Natural Language Processing (NLP) and text mining facilitate the analysis of natural language text. The text is first tokenised for processing unstructured clinical text with NLP (the text is divided into words and delimiters). Next, the statistical properties of the text can be analysed by classifying words into adjectives, verbs, nouns, etc. Words are further divided by statistical significance and syntactical features [21]. Preprocessing the text in these steps transforms unstructured clinical text into structured data, which can later be used as training data in machine learning [1].

Another type of unstructured EHRs is medical images, either static images (X-rays, computed tomography, ultrasound, etc.) or recordings (surgeries). Deep learning helps doctors interpret medical images by denoising the images [22]. Other data analyses improve CDS by classifying or detecting lesions on medical images [23].

Some challenges associated with analysing and using the data in clinical trials include data access, privacy, quality, and validation. Accessing the EHRs is complicated due to the sensitive nature of the data and the lack of facilities that provide secondary access to data. One of the ways to address this is deidentification and pseudonymisation of the data [24]. Surrogates or pseudonyms can replace patients' identification. Another problem stems from this approach: replacing personal identification in unstructured data (e.g. clinical text). First, the personal information must be found in the clinical text, and then deidentification can be performed. Analysis of unstructured data is not straightforward, but computational methods for finding the patient’s identifiers in data have proved successful in this task [25]. Another approach is to adapt analysing techniques to preserve the patient's privacy during computation, for example, by complementing machine learning techniques with privacy-preserving features. This is achieved when data are distributed and partially analysed across data owners, while only aggregated results and statistics are revealed to outsiders [26].

Complete data capture, quality, and validation are challenges that must be addressed in the EHR analysis and for using EHR data for clinical studies. The data quality assessment framework should involve checks for completeness of the data and plausibility [27]. Moreover, data must be mapped using ontologies (e.g. ICD-10 codes) and standard medical terminology [28]. On condition that the limitations of EHRs are understood and considered, real-world patient data can be used for CDS and scientific research.

Nowadays, one of the common approaches to connect EHRs with general healthcare and genetic research is the creation of biobanks such as the Estonian Biobank. Biobanks enable scientists to analyse real-time medical data to promote personalised medicine and biologically informed healthcare [29].

Progressive views of the Estonian government towards science, specifically personalised medicine, have created a favourable environment for the development of biobanking initiatives. In 2000, the government passed the Human Genes Research Act, which gave a legal basis to the Estonian Biobank [30]. Furthermore, Estonia invested in developing a massive digital infrastructure, which aided in the development of Estonian EHRs and accelerated the sharing of medical data between doctors, patients, and hospitals [31]. All these factors culminated in the Estonian Biobank (EstBB) establishing in 2001. Participants of the EstBB volunteered to provide their data for scientific research by donating their blood samples and filling out questionnaires about their lifestyles and clinical diagnoses in structured data format. Additionally, whole genome sequencing, transcriptomics, different metabolic profiles, and proteomics were performed for the part of the cohort [32]. The EstBB contains data from over 200,000 individuals, constituting 20% of the Estonian population [33].

1.2 Multimorbidity and comorbidity

As a consequence of the implementation of EHRs, an extensive amount of data has become available for scientists and laid a foundation for studying diseases from different points of view. EHRs facilitated the analysis of diseases not only as static sets of symptoms but also as chronological trajectories representing how diseases interact and transform into one another. From the big data, comorbidity networks have been constructed to identify disease interactions and disease trajectories to illustrate transformation events [34]. Comorbidity can be defined as the presence of another disease with reference to the disease being treated [35]. The presence of two or more long-term physical or mental health conditions in the same individual is called multimorbidity [36].

The occurrence of two or more diseases in one patient has become a persistent fact due to the same underlying cause of multiple diagnoses, the increasing life expectancy of humans, and

external factors. For instance, elevated inflammatory markers are associated with the co-occurrence of chronic diseases [37]. A study performed by Fraser *et al*. investigated the contribution of biological hallmarks of ageing to the development of age-related diseases and multimorbidity. An exploration of EHRs established that the five hallmarks of ageing were coupled more frequently with the co-occurrence of age-related diseases than would be expected by chance [38]. Furthermore, multimorbidity is frequent in low- and middle-income countries. In such regions, it appears at earlier ages due to unsatisfactory environmental conditions, lack of healthcare infrastructure and poverty [39].

Both multimorbidity and comorbidity reduce a patient's quality of life [40] and burden healthcare by increasing healthcare utilisation and costs [41]. Complexity to treating multimorbidity is added because current medical protocols focus on treating one condition at a time. Therefore, managing multimorbid patients favours more critical conditions and milder ones may be overlooked. Hence, there is a call for creating separate approaches for treating comorbidities by adding cross-references in the medical guidelines, adding older people and people with comorbidities into clinical trials, etc. [42]. There is no standard for addressing multimorbidity in patients, but several international generic guidelines exist for approaching this condition [43, 44].

Recognising co-occurrence and interaction of diseases leads to the belief that diseases should not be studied separately. As previously stated, comorbidity networks and disease trajectories have been introduced by scientists to study mechanisms and pathology of diseases and multimorbidity. The next chapter describes several disease trajectories identification studies and how researchers investigate the co-occurrence of diseases. However, disease trajectories are not yet coupled with general healthcare and cannot be utilised by doctors to diagnose and treat individual patients. Doctors should be able to view all patient medical data concurrently through abstraction and visualisation tools to improve CDS when dealing with multimorbidity and comorbidity.

1.2.1 Studying multimorbidity with disease trajectories – scientific point of view

The concept of disease trajectories has been developed to explore the progression of multimorbidity. Disease trajectories showcase the diagnoses that occurred one after the other in a specified time frame in many patients. By studying EHRs and creating large cohorts of patients, scientists can make statistically significant progression timelines of disease changes. These paths illustrate which symptoms or diagnoses are associated with the disease and how metabolic markers (analyte concentrations) assessed by medical laboratory observations change throughout the disease. Doctors are interested in seeing these paths to know how far their

individual patients are from the illness, at which point of the disease they are, and predict what the next event can be.

Only a few large-scale population studies on identifying disease trajectories in populations have been conducted so far. One of the first studies was done by Jensen *et al*. and is based on the Danish population's EHRs stored in The Danish National Patient Registry (NPR) [45]. Extracted data from NPR were used in this study to identify disease trajectories. Data covered three types of interactions with hospitals of the Danish population: inpatient admissions, outpatient admissions, and emergency room visits. Data spanned 14.9 years, yielding 6.2 million patients with 65 million healthcare encounters. Encounters were encoded in the ICD-10 system and produced 101 million unique diagnoses. The general workflow was first to identify diagnosis pairs and then cluster them in longer paths.

For identifying pairs of diagnoses where the first diagnosis, D1, occurs before the second diagnosis, D2, in 5 years (temporal correlation), the relative risk (RR) measure was used. More than a million pairs were identified, but some ICD-10 codes were removed (e.g., related to pregnancy, general symptoms, or caused by external causes). From the remaining codes, ~70,000 were encountered more than ten times, had RR>1, and were statistically significant. Additionally, the pairs were tested for directionality, meaning that only pairs with a significantly higher count of patients had the D2 occurring after D1 and not the other way around. The final amount of ~4,000 diagnosis pairs were clustered into longer trajectories. Later, the study was expanded by including more NPR data and creating the Danish Disease Trajectory Browser [46, 47]. This browser allows anyone to search, inspect, analyse, and filter disease trajectories from the Danish population.

Another important example of a study was conducted to identify disease trajectories in the Spanish health registry on male prostate cancer patients. It not only identified temporal disease trajectories but connected them with phenotypic and genetic data. The study workflow started with identifying shared disease sequences between patients of length two or longer. Then statistically significant diagnosis pairs were identified, RR score across different age groups was used to finalise the list of pairs, and only pairs with the preferred directionality were used for clustering. The clustering was done using three disease similarity metrics: clinical (comparing relatedness in meaning of diagnoses codes), genetic (comparing by identifying similarities between diseases on gene level), and phenotypic (comparing by phenotypes of diseases). This study resulted in disease trajectories represented in different spaces, allowing an understanding of disease development from various levels [35].

Dervić *et al.* conducted a large-scale population study on the data from patient in-hospital stays in Austria spanning 17 years. Data were separated into 10-year age groups. Therefore, disease trajectories have two links: intralayer (links that connect diagnoses in the same age group) and interlayer (links that connect diagnoses from different age groups). All significant correlations of the diagnoses in the same group were identified as intralayer connections, and correlations in diagnoses from various age groups were identified as interlayer connections. Then, a community detection algorithm was used on the detected comorbidity networks to identify disease trajectories. In this study, the trajectories were considered communities of diagnoses rather than exclusive non-overlapping clusters, allowing the common nodes between disease trajectories to be considered. Identified disease trajectories were used to locate critical events – bifurcation points in disease trajectories that precede diagnoses of higher mortality or prolonged hospital stays [34].

Disease trajectories emerged to study multimorbidity and development of the disease. With the advances in personalised medicine, individual patients' data will be compared against many trajectories common to the population in question. Such comparisons should create a risk assessment of developing a medical condition for the patients and promote preventive healthcare and timely treatment.

1.3 Visualisation of EHRs – doctors' point of view

Tools for identifying disease trajectories are a significant step towards personalised medicine and the improvement of CDS. Likewise, they are a way to study disease from a scientific point of view. Doctors can approach multimorbidity in their patients by analysing a comprehensive visual representation of the patient's anamnesis, which must include multiple data types. There is a need for tools that provide a clear, aligned visualisation of longitudinal medical data. Such software should visualise all medical records simultaneously to assess and diagnose the patient's condition.

Dabek *et al*. proposed a framework summarising patient EHRs into one timeline with summary nodes corresponding to dates. Visual representation of a node shows what data are available for that date. The node can consist of several elements, and by altering the size and order of elements, the importance of the information encoded can be communicated to a user [48].

HARVEST is a tool which focuses on extracting information from clinical notes. NLP processes clinical notes, and then HARVEST produces the patient's timeline and a problem cloud. The patient's timeline summarises the patient's visits, and the cloud showcases the most important diagnoses and symptoms extracted from the data. The user can view raw notes connected with the element selected by clicking on the cloud elements [49].

There were also tools developed for a specific application. Medical Information Visualisation Assistant (MIVA) was designed as a CDS for doctors in intensive care units [50]. Steinhauer *et al.*, developed a tool for Dermatological Oncological Tumour Board [51]. It visualises all the relevant information about the cancer treatment of the patient and their effects on the patient to support the selection of suitable subsequent treatment. Zhang *et al.* created IDMVis to visualise events in type I diabetes treatment and adjust the treatment [52].

The most recent tools developed for concurrent visualisation of several medical data types are PatientExploreE, LifeTrack, and ClinicalPath. PatientExploreR is an open-source visualising tool based on R and shiny [53]. This web-based application works with the Observational Medical Outcomes Partnership (OMOP) Common Data Model (CDM) introduced by Observational Health Data Sciences and Informatics (OHDSI) [54]. OMOP CDM is a data standard that converts observational data from databases into common terminologies, coding schemes, and vocabularies. These data can be used for the analysis and to produce trustworthy evidence. According to the OHDSI 2023 report, 331 databases from 34 countries have already adopted the OMOP CDM, resulting in 810 million patient records.

PatientExploreR allows the user to visualise data from records to which the user has authorised access. Users can search the database through the application to select specific patients by ID or to select a cohort with the same condition. If visualising an individual patient, a clinical summary of the patient is displayed, which shows how many entries of what type are recorded for this patient and the patient's background. If a cohort of patients is studied, then the statistics of the cohort will be displayed, such as counts of male and female patients, nationalities and races of patients in the cohort, and age distribution. From the cohort, individual patients can then be visualised. The report in .csv format can be exported for further use in other software [53].

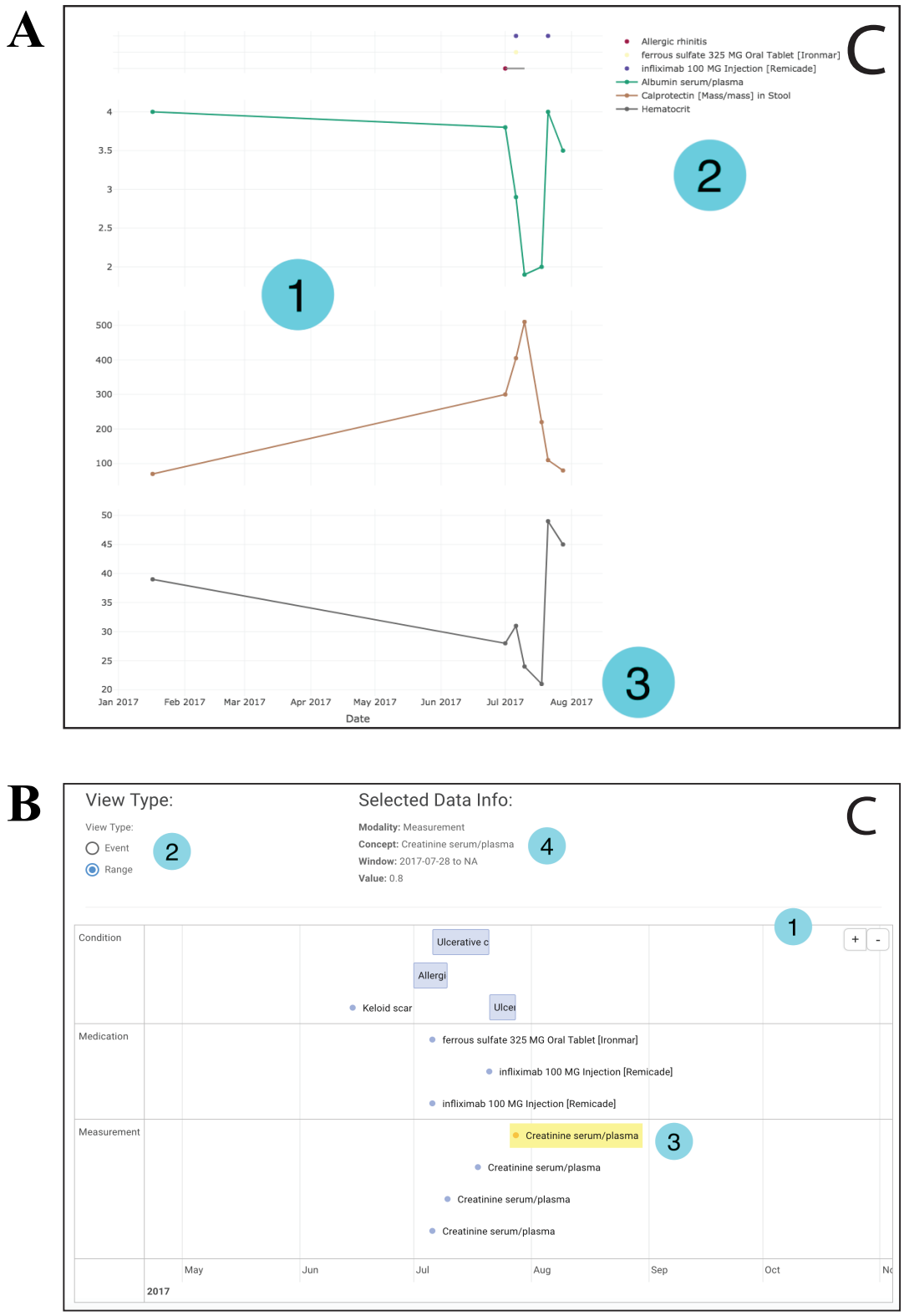

Fig. 2. A) Multiplex mode; 1. Plot showing Condition and Measurement event types; 2. Legend with items selected by a user; 3. X-axis with time. **B)** Multiplex timeline mode; 1. Plot showing Condition, Medication, and Measurement event types; 2. Menu for selecting how to view data; 3. Selected by click event; 4. Information for the event chosen [53].

The application has a Data Explorer section which allows the inspection of categorical and numerical data in 3 modes: targeted (one event type at a time, e.g., only laboratory measurements), multiplex (multiple event types plotted over time, Fig. 2A), and multiplex timeline (multiple event types grouped and plotted over time, Fig. 2B).

Additionally, there is a website, PatientExploreR, based on synthesised patient data [55]. This website was created to let users without official access to an EHR database get to know the software and learn how to use it without using any sensitive medical information.

Limitations of PatientExploreR are data format, limited access to the application, and lack of minor functionalities. OMOP CDM is a strict data format, and as the authors of this software note, converting data into OMOP CDM format is strenuous, requires trained personnel, and takes considerable time. The user must have authorised access to an EHR database to analyse data with this software. Furthermore, some plots from the application cannot be saved or have poor resolution.

In 2023, two prominent EHR visualising tools were published: LifeTrack and ClinicalPath [56, 57]. LifeTrack is an application which uses raw register data of a cohort. The foundation of this tool is the “overview and detail” paradigm. The user always has an overview of the data and can access more details by hovering over the data (Fig. 3). Firstly, the user uploads a file with the cohort's data, then selects individual patient ID, and the application visualises the patient’s EHRs as scatter plots. The event types displayed can be the reason for hospital visits, procedures, and ICD diagnoses. According to the authors, LifeTrack is already used by clinicians to familiarise themselves with the records of a new patient in a short time. However, no case study has been shown to confirm this.

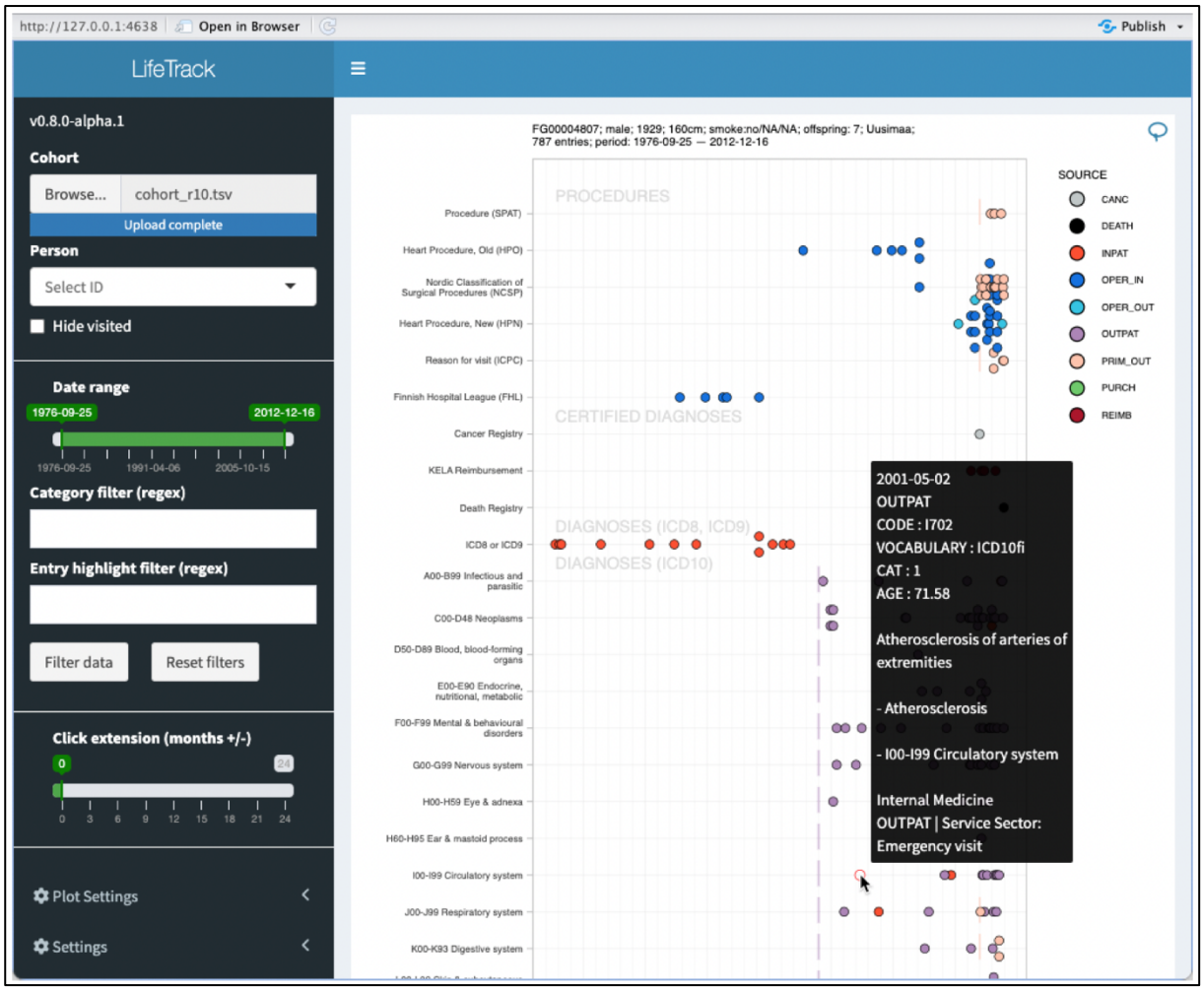

Fig. 3. User interface of LifeTrack tool. On the left are visualisation controls, which include a cohort, the bar for selecting the patient's ID, a selection of time period, and filters to select specific conditions or entries in the data. On the right side, there is a plot with visualised EHRs. The patient's background information is at the top of the plot. The Y-axis represents classes of entries. Data points are colour-coded based on the source of the registry, e.g., the Finnish Cancer Registry denoted as CANC. By hovering over the plot data points, more information can be viewed [56].

Limitations of LifeTrack are limited data types that can be displayed and restricted access to the application. Since the tool is still under development, it cannot display laboratory measurements and clinical notes. Additionally, this tool is part of the FinnGen research project coupled with the Finnish Biobank [58]. Therefore, the user must have specific access to FinnGen data to use the software.

ClinicalPath is a tool that visualises laboratory measurements by creating a map of symbols encoded by shape and colour. It was developed based on the opinions of domain experts, who identified strict requirements for the application before the development.

Coding in ClinicalPath is redundant: the colour and shape of the symbol representing laboratory measurement have the same meaning (Fig. 4C). Some examples of the symbols are a green

rhombus representing a normal test result, a blue triangle pointing downwards representing a result below reference value, and a red triangle pointing upwards representing a result above reference value. Tests are plotted chronologically, and dates on the X-axis are colour-coded based on the reason for the patient's visit (Fig. 4B). Moreover, test results can be visualised as line charts to track exact changes in test results (Fig. 4D and 4E). The tool can highlight the relevant changes in test results (Fig. 4F). Users can filter the dates and tests to be displayed, reorder them, and zoom in on specific time periods.

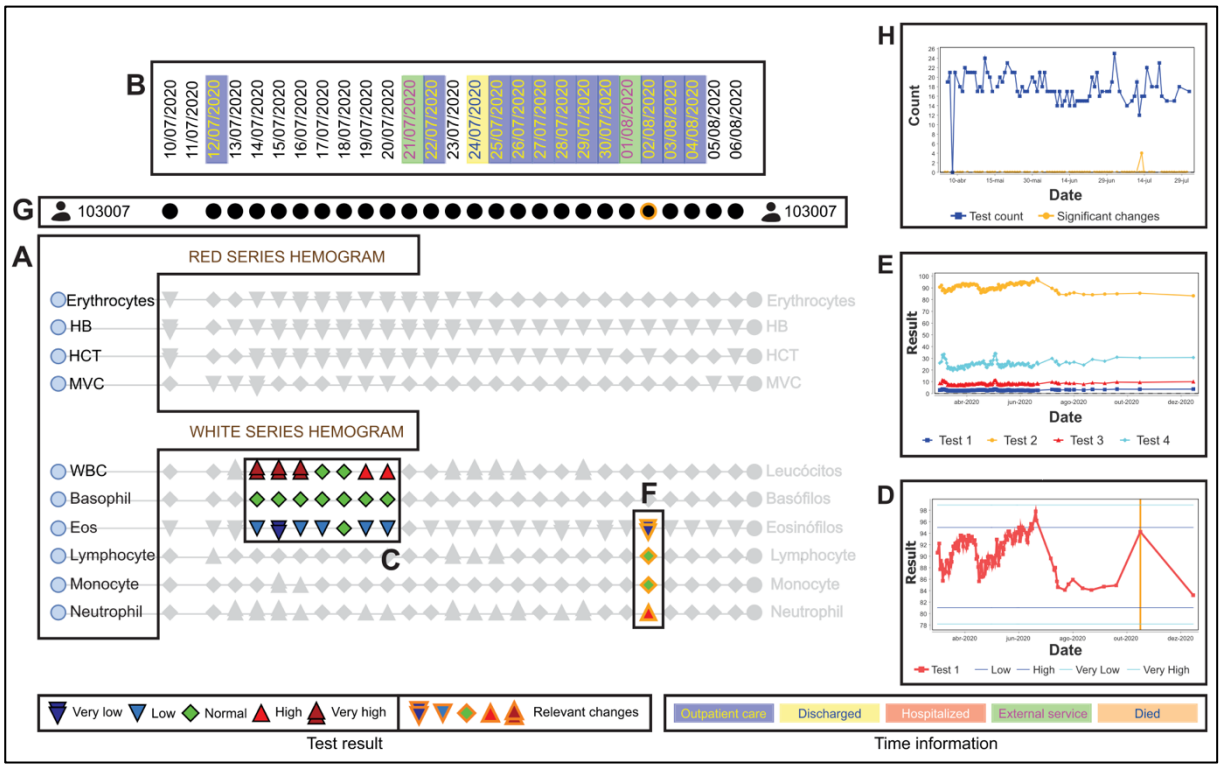

Fig. 4. ClinicalPath tool EHR visualisation example. **A)** Tests' categories. **B)** Timestamps colour-coded by the type of healthcare encounter. **C)** Colourful shapes encoding measurement's result. **D)** Line chart with measurements changes over time. **E)** Line chart combining several measurements. **F)** Highlighted changes in test results. **G)** Patient information over time. **H)** Line chart with the test counts over time [57].

The authors of ClinicalPath also conducted user evaluation experiments. They recruited 15 doctors who were provided with the tool installation guide and instructions and had to fill out a questionnaire. The questionnaire contained questions to test whether the user understood the tool's functionality and could diagnose the patient using the tool and additional patient background. The group of doctors achieved 90.4% of correct answers, and the application received positive feedback [57].

Some of the limitations of the application highlighted by the authors are filtering of the laboratory analysis names and lack of functionality to display other types of data. The authors

had to filter out some test types due to differences in notations between hospitals, errors in the labels, various scales of tests, and analyte names. Hence, some of the tests were excluded from the visualisation, and the tool was tested on the preprocessed data different from the clinical setting. Furthermore, the tool cannot display medical images, which also aid in diagnosing patients, and adding them to the tool is one of the future directions.

Table 2. Comparison table of the most recent EHRs visualisation tools.

Tool's name	PatientExploreR [53]	LifeTrack [56]	ClinicalPath [57]
Programming language	R	R	JAVA
Data type used	OMOP CDM	Raw registry data	.txt files
Code availability	Open-source	Open-source	Open-source
Types of data visualised	Laboratory analyses, diagnoses; medications; types of hospital encounter; procedures; observations; medical devices.	Diagnoses; medications, types of hospital encounter; procedures.	Laboratory analyses; types of hospital encounter.
Allows concurrent visualisation	Yes	Yes	Yes
Limitations	Used only with authorised access to an EHRs database; strict data format.	No laboratory measurements and clinical notes; access only through the FinnGen database.	Limited data types visualised; reduced number of laboratory measurements visualised.

A comparison table of the most recent EHRs visualising tools is presented in Table 2. Every software is unique, and its usefulness cannot be understated. All listed tools are different because they are tailored to the specific needs of the people who created them. Even though the design features of these applications were called limitations, some are also strengths. For example, allowing the use of the software only in conjunction with authorised access to the EHR database guarantees the security of sensitive medical data and patient privacy. Hence, the diversity of design features in tools for visualising medical data highlights the needs of doctors and scientists for such software and the many ways it can contribute to diagnosing patients or studying diseases.

2 AIMS OF THE THESIS

This Medi_Vizz project originated at the request of the Institute of Genomics, University of Tartu researchers, who needed an application for concurrent visualisation of medical data for individual patients. Hence, the aims of this thesis are:

- Develop a tool enabling scientists to visualise the disease trajectory of individual patients composed of the diagnoses encoded by ICD-10 codes and laboratory measurements.
- Aid doctors in diagnosing patients and dealing with multimorbidity by encoding several layers of information in one representation.

3 MATERIALS AND METHODS

This chapter describes the tools, libraries and the environment used for developing the Medi_Vizz tool, the application's code structure and the testing dataset.

3.1 Libraries and Development Environment

Libraries

- Plotly 5.19.0 and Dash 2.14.2

Medi_Vizz application was produced using Python, and Plotly and Dash were used for web development. They are libraries for data scientists and engineers who want to put their analytics or modelling code into an application to visualise data. Medi_Vizz produced a timeline consisting of two interconnected plots: the top displays diagnoses encoded by the ICD-10, and the bottom presents laboratory measurements. Plots were created with the help of the Plotly open-source library. Diagnoses can be visualised in two ways: each with a separate bubble (bubble plot) or grouped and plotted as lines (lines plot). The bottom panel is a ribbon plot which displayed laboratory measurements numerical results. Dash is used to make the layout of the tool and user interface using Dash components (dropdown menus, buttons, graphs, etc.). [59].

- NumPy 1.26.4

NumPy is an open-source Python library which allows numerical computing. In Medi_Vizz, this library was used to read data from the files and preserve data when transferring data between scripts [60].

- Kaleido 0.2.1

Kaleido is a library for static images (e.g., .svg or .pdf) developed for the web visualisation library Plotly. This library was used to save the plot in .svg format [61].

Additionally, the Dash Bootstrap components library version 1.5.0 was used to style the buttons [62]. Pandas version 2.2.1 and dateutil module version 2.8.2 were used to handle time data. Pandas is a data-analysis library, and dateutil is an extension of the datetime module by default available in Python [63, 64]. Webbrowser Python module was used to run the application in the web browser. This library gives a simple way to open the link to the application in the default browser without dependence on the operating system [65].

Development environment

The computer on which the Medi_Vizz was developed has the following hardware specifications:

- Processor: Apple M1 Chip
- Processor's speed: 3.2 GHz
- RAM: 16 GB

The operating system used on the development computer was macOS Sonoma, version 14.4.1. The application was developed in Visual Studio Code editor, which has many extensions, allowing to write code in different programming languages, including Python [66]. The version of Python used for writing the tool was 3.12.2. The application has a user interface which is accessed from the web browser. For this, the default MacOS browser Safari Version 17.4.1 was used.

3.2 Medi_Vizz code architecture

Dash_app.py is the main script which starts the server on which the Dash application runs. This script specifies the webpage layout using Dash components. Likewise, it is responsible for interaction with the user, and it calls helper scripts to process the uploaded data and the user's requests. In the dash_app.py script, all menu components with callbacks are defined. Other scrips are utilised based on the user's interaction with the elements. Callback graph of the Medi_Vizz can be accessed when debugging the Dash application (Fig. 5).

The dash_app.py script utilises two programs, subplots_dt1_labres.py and subplots_dt2_labres.py, to create a panel with subplots: diagnosis codes plot and laboratory measurements plot (Fig. 6). These scripts combine the two data types in one visualisation and send the plot to the dash_app.py. Furthermore, these scripts save data into a .svg picture and a .txt file.

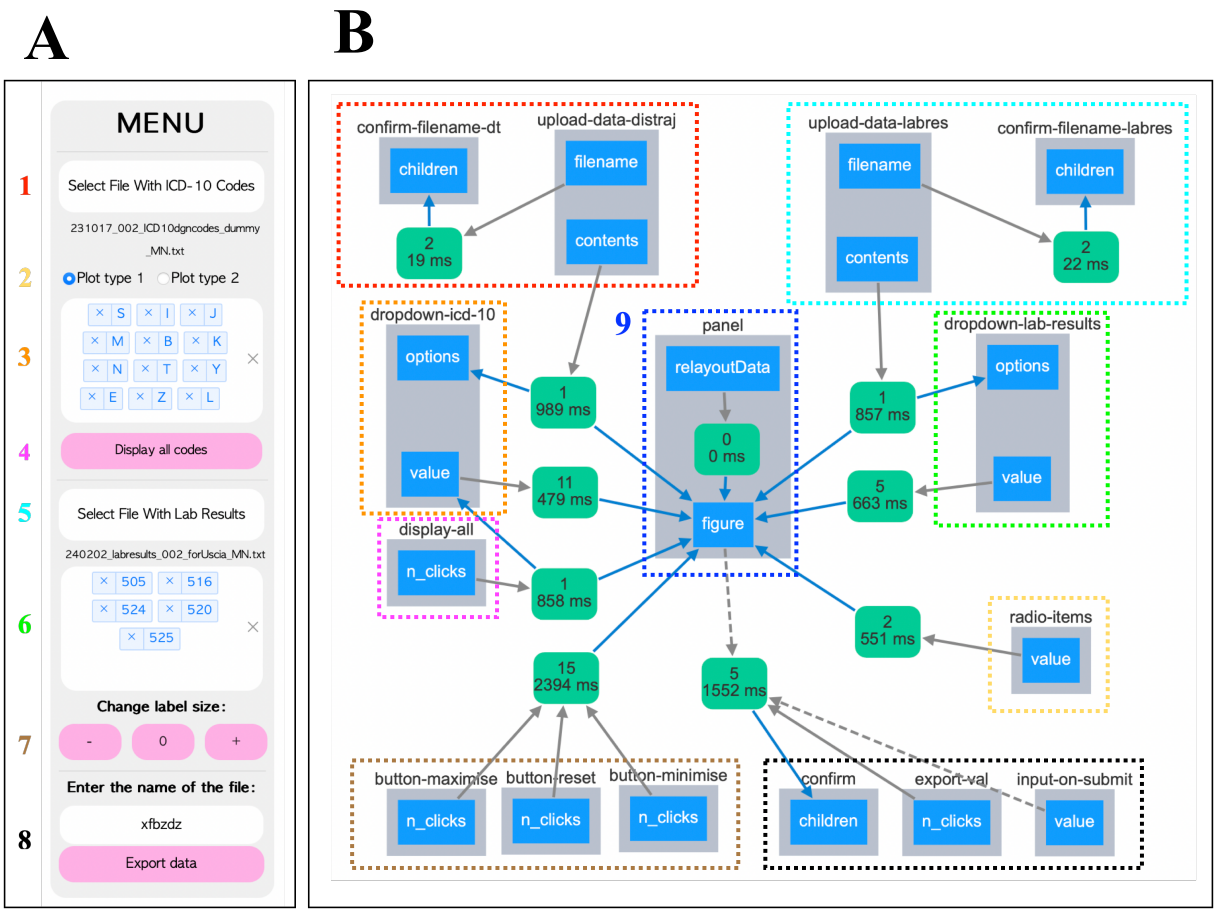

Fig. 5. A) Menu components numbered and colour-coded (can be viewed and controlled by the user). 1. Box to upload a file with ICD-10 codes; 2. Buttons to switch between two ICD-10 type plots; 3. The dropdown menu to select which ICD-10 chapters' codes will be displayed on the plot; 4. Button to display all diagnosis codes at once; 5. Box to upload a file with laboratory measurements; 6. The dropdown menu to select which laboratory measurements will be displayed on the plot; 7. Buttons to change the size of labels on the plot; 8. Box to enter the name of the exported .svg picture and .txt file; **B)** Callback graph of the Medi_Vizz application (not accessible to the user but provided by Dash library during application development). The corresponding colour box highlights the callback of the menu component. Grey boxes show components' callbacks (e.g., of the main plot, dropdown menus or buttons), blue boxes show what data is utilised in the callback (e.g. "value" of the "dropdown-icd-10" menu is a list of the options selected), arrows represent the communication of the components. The green boxes' upper line shows the number of times the callback was used, and the bottom line shows the timestamps in milliseconds, which indicates the order in which callbacks were executed. 9. Main visualisation callback, modified by graphical interface components.

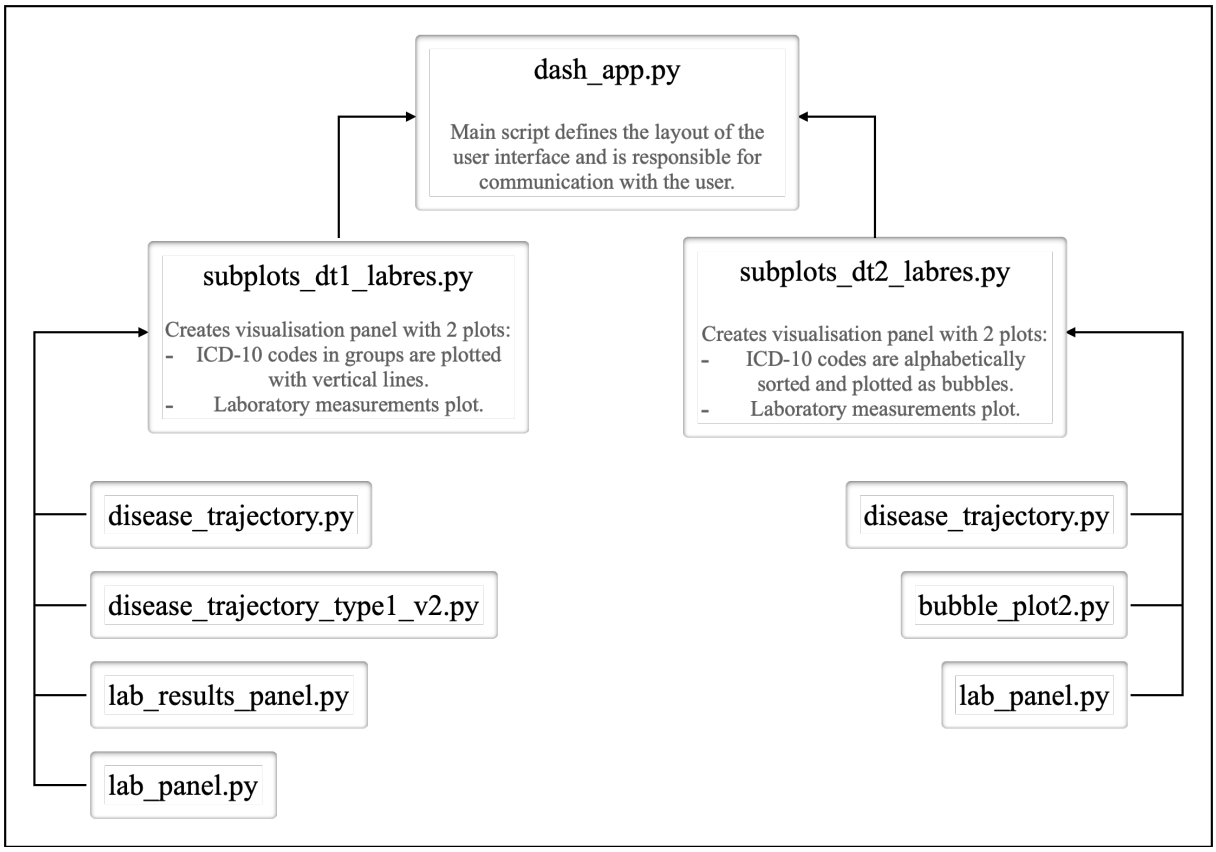

Fig. 6. Medi_Vizz code structure and communication between scripts. Arrows represent the import statements (e.g. dash_app.py imports scripts subplots_dt1_labres.py and subplots_dt2_labres.py).

ICD-10 visualisation plots, bubble plot (Fig. 7A) and lines plot (Fig. 7B), are encoded by either bubble_plot2.py (codes sorted alphabetically and plotted as bubbles on a string) or disease_trajectory_type1_v2.py (codes sorted based on the number of codes assigned in one day and plotted as lines). The laboratory measurements panel is created using lab_panel.py and lab_results_panel.py (Fig. 7C). Scripts responsible for producing plots read the required data type from the uploaded file, sort data, and generate plots. Lab_panel.py script receives a subset of data which was extracted from the uploaded file based on the name of the test. After that, the program determines the lowest and highest measurement values and distributes the heights of all the other results between the minimum and maximum values.

In case of the bubble plot, the frequency of the codes from one ICD-10 chapter is considered. More space on the plot is allocated to the chapter with higher counts (Fig. 7A). Codes from the same day will be connected with one line. The thickness of the line is calculated based on the number of codes in one day. On the line, the bubbles are plotted and sorted alphabetically. In the case of the lines plot, codes assigned to a patient on the same day are displayed in groups. The group's height is based on the size of the groups: the more codes the group includes, the further from the main line this group is plotted (Fig. 7B).

For ICD-10 timelines, zooming in on a specific time period or choosing ICD-10 codes starting letters from the dropdown menu is equivalent to selecting a subset of data. For the bubble plot, the codes' frequency will be calculated in the specified time period, and the space will be redistributed again. For the lines plot, the lines are redistributed again based on the length of the group of codes.

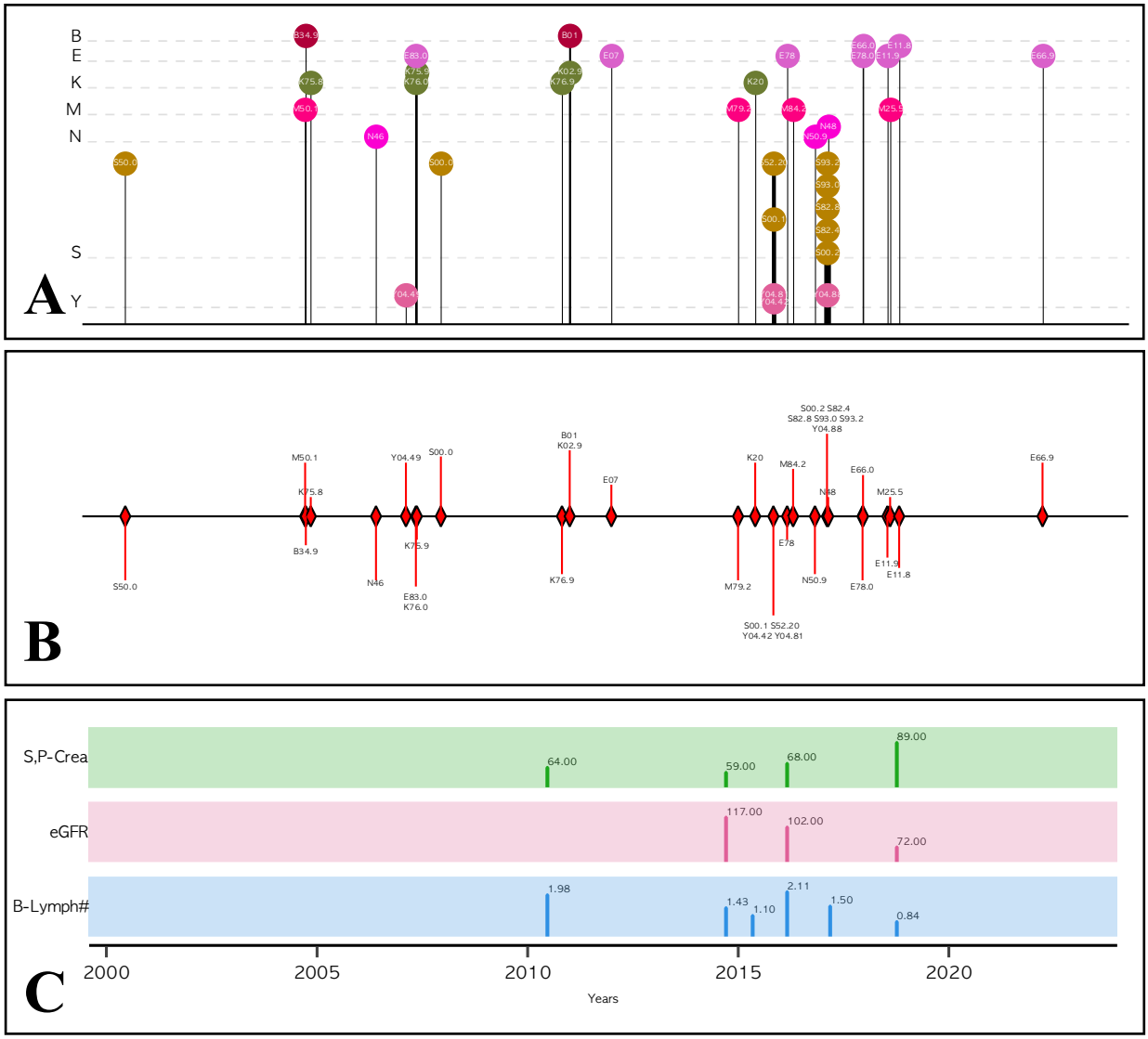

Fig. 7. Types of plots that the Medi_Vizz tool can generate. **A)** Bubble plot with diagnoses encoded by ICD-10. **B)** Lines plot with diagnoses encoded by ICD-10. **C)** Laboratory measurements plot.

3.3 Dataset

Pseudonymised data from the EstBB was used to test the application. These were structured data of two types: diagnoses encoded by the ICD-10 and laboratory measurements. Data uploaded into Medi_Vizz are tab-delimited tabular files. In the file with ICD-10 codes, one

column must contain dates, and the other has to contain codes. In the file with laboratory measurements, three columns are required: test dates, analysis names and numerical results.

Data for EstBB is collected according to the Estonian Human Genes Research Act. Volunteers recruited through advertisements, general practitioners or physicians in Estonian hospitals provided the researchers consent to use their EHRs and genomics data. After signing the consent forms, volunteers completed questionnaires about their lifestyle, health, occupation, and family history. Researchers can contact the EstBB participants to conduct further studies [32]. Moreover, EstBB is connected to national registries (e.g., hospital EHRs, Cancer Registry, National Insurance Fund, etc.). The current number of participants is more than 200,000, representing about 20% of the adult Estonian population [67].

The dataset used for testing the application did not contain personal information and could not be traced to any individuals. It was artificially generated to resemble the data from the EstBB and provided for this project by the Institute of Genomics, University of Tartu.

4 RESULTS AND DISCUSSION

This chapter describes the Medi_Vizz application layout and functionalities, discusses the limitations and considerations, and presents future work on the application and its place in CDS.

4.1 Graphical Interface

Medi_Vizz runs through a default browser of a computer, e.g., Safari browser (Fig. 8).

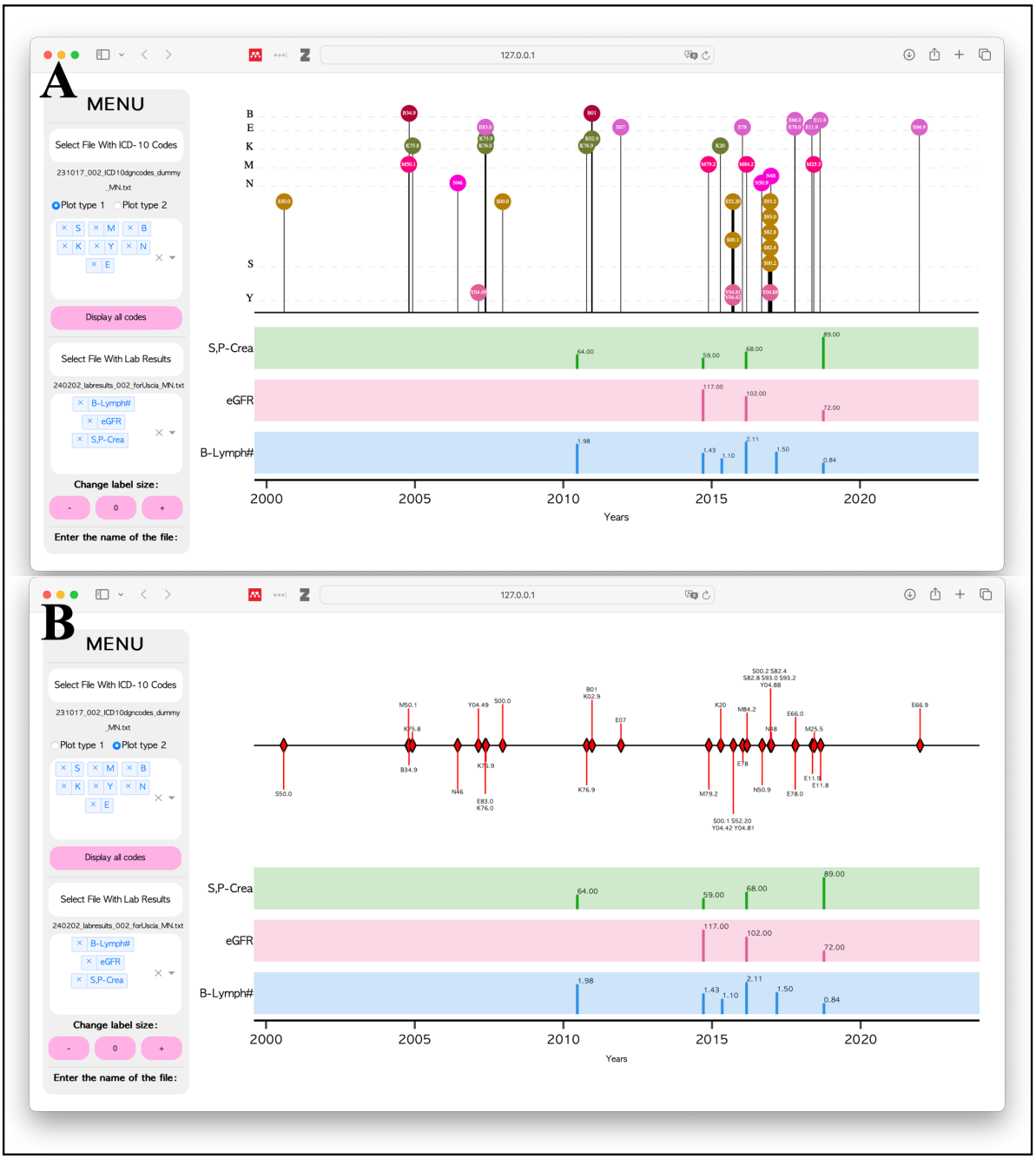

Fig. 8. Interface of the Medi_Vizz tool displayed in the Safari browser. On the left side, the menu to control the plot is shown; on the right side, the resulting plot is displayed. **A)** Panel with ICD-10 diagnoses displayed as bubble plot and laboratory measurements. **B)** Panel with ICD-10 diagnoses displayed as lines plot and laboratory measurements.

The menu that is used to control the layout is on the left side. On the right side, the resulting plot is displayed. Diagnoses encoded by the ICD-10 system are plotted on the upper half of the plot, and ICD-10 chapters are B, E, K, M, N, S, and Y. Users can select which chapters to plot using the searchable dropdown list from the menu and switch between two types of ICD-10 codes visualisation using menu buttons “Plot type 1” (Bubble plot, Fig. 8A) and “Plot type 2” (Line plot, Fig. 8B). Bubble plot sorts the diagnoses alphabetically and the thickness correlates with the number of diagnoses assigned in one day. Line plot positions the diagnoses based on the number of diagnoses assigned in one day. By pressing “Display all codes”, every available diagnosis code of the patient is displayed. The lower part of the plot shows the laboratory measurements plotted as blocks. Users can select which measurements to plot using the searchable dropdown list from the menu. For instance, analyses S,P-Crea – creatinine measurement in blood, eGFR – glomerular filtration, and B-Lymph# – number of lymphocytes are displayed (Fig. 8).

Users can control the size of the labels and zoom in on specific time periods using the range selector on the plot (Fig. 9A). Additionally, by hovering over the points, the exact date of the diagnosis or test is shown (Fig. 9B, Fig. 9C). The displayed plots and data can be exported to a .svg file and a .txt table using the “Export data” button (Fig. 5-9).

Medi_Vizz can be downloaded by anyone from the GitHub repository https://github.com/PlanetWyh/Medi_Vizz. The repository contains all the code along with the user guide. The user guide describes installing Python and the libraries required to run the application on macOS, Linux, and Windows. It also includes a description of the tool’s functionalities and a link to a short example video. Medi_Vizz starts from the terminal (macOS/Linux) or command line (Windows). After the start, the default browser opens with the application.

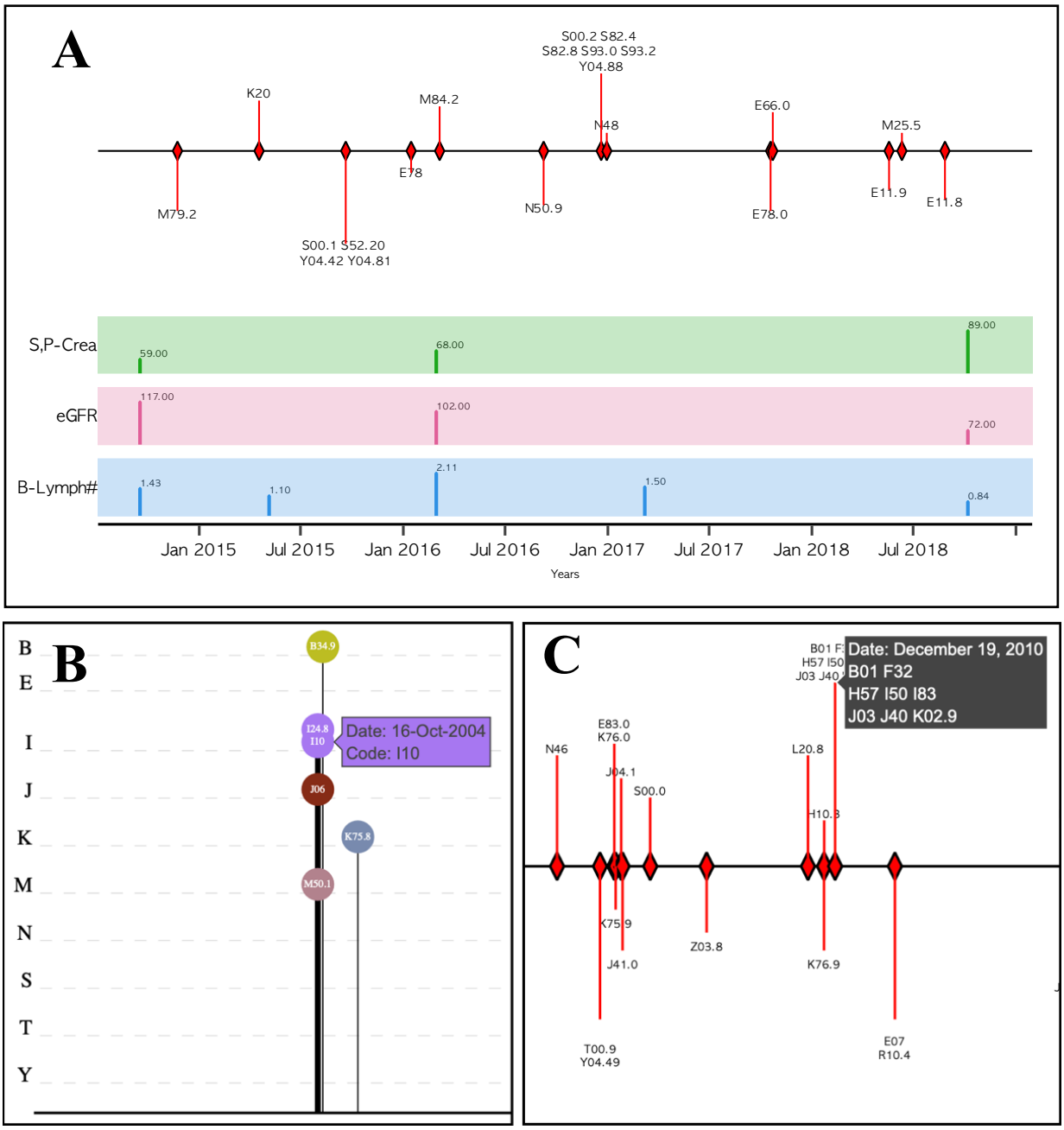

Fig. 9. A) Zoomed plot from 2014 to 2018 with the increased label size. **B)** Window shown when hovering over a bubble with diagnosis code I10. **C)** Window shown when hovering over a line with a group of diagnoses in lines plot.

4.2 Limitations and considerations

Medi_Vizz tool meets all the requirements set by the goals of this thesis. However, there are a few things to consider and highlight for future improvements. Medi_Vizz is limited to two data types: ICD-10 diagnoses and laboratory measurements. As an improvement, several data types could be added to represent the patient's medical history. Examples can be treatments that patients underwent, types of hospital encounters, and answers to binary questions. Medi_Vizz does not provide data validation, and the user is responsible for ensuring that data are correct before uploading it. For example, if the user uploads the laboratory measurement with a

negative numerical result, Medi_Vizz will not consider this an irrelevant value and will not provide an error message. Another case is using different units for the same laboratory test. Medi_Vizz does not convert the units and will display such measurements as two separate tests if the names of tests differ or as one test with huge variations if the name of the test is the same. Furthermore, this tool is not designed with data protection issues in mind. It runs without an internet connection, but users must be aware not to upload personal identification or sensitive information with their data.

4.3 Discussion

Medi_Vizz is a tool for concurrent visualisation of structured medical data of two types: ICD-10 diagnoses and laboratory measurements. It was developed and adjusted to the specific needs of the Institute of Genomics, University of Tartu researchers, but can be used by anyone who needs to visualise medical data. Medi_Vizz allows scientists to study the disease and medical test trajectories of individual patients and for doctors to diagnose their patients and deal with multimorbidity.

The future work should include the addition of other data types and a case study with doctors and scientists. The case study will assess how convenient and comfortable it is for potential users to use Medi_Vizz. Medi_Vizz was already tried out by one researcher at the Institute of Genomics, University of Tartu (Appendix, chapter 2). However, the case study should have more people and an in-depth questionnaire. Users' answers should be implemented in the design of the user interface, and new functionality which users lack can be introduced. As a long-term goal, Medi_Vizz and visualisation tools reviewed in this thesis should be used in conjunction with disease trajectories studies and browsers. Allowing the doctors to compare their patient's anamnesis to the disease trajectory can help predict disease development early on. This could contribute significantly to developing precision medicine and disease prevention since doctors will be informed about potential conditions their patients can develop based on the earliest symptoms.

SUMMARY

Doctors need to evaluate the information available for one patient for accurate diagnosis and treatment prescription, which is particularly complicated for patients with multiple chronic conditions treated by different specialists. With paper-based medical records, access to information was fragmented, and doctors could not assess entire patient history. The emergence of EHRs granted doctors a cohesive view of patients' medical data from their workspace, introducing awareness in the diagnosing process and enhancing its accuracy. Nevertheless, the representation of information clearly and comprehensibly remained a challenge. Current applications developed for visualising medical data often require authorised EHRs database access or are limited to specific needs or data types.

The Medi_Vizz tool was developed to address these issues and allows the assessment of two types of medical data of individual patients: diagnoses and laboratory analysis. This tool has a user-friendly interface and intuitively visualises medical data. Medi_Vizz is available to everyone from the online repository to support the application's future development and promote its customisation. The repository contains detailed user guides for three operating systems to make the tool accessible to everyone. An MD employed by the University of Tartu Institute of Genomics tested the application using the repository and user guides and responded with positive feedback (Appendix, chapter 2).

The Medi_Vizz application facilitates clear and effective visualisation of medical data and contributes to general healthcare and clinical research. Plots produced by Medi_Vizz can be utilised for CDS for doctors and help scientists study the EHRs of individual patients and disease progression. With future work, Medi_Vizz can be used with prediction models and disease trajectories browsers to produce a risk assessment for individual patients and contribute to preventive healthcare.

REFERENCES

[1] M. Tayefi *et al.*, 'Challenges and opportunities beyond structured data in analysis of electronic health records', *Wiley Interdiscip Rev Comput Stat*, vol. 13, no. 6, p. e1549, Nov. 2021, doi: 10.1002/WICS.1549.

[2] C. L. Camp, R. L. Smoot, T. N. Kolettis, C. B. Groenewald, S. M. Greenlee, and D. R. Farley, 'Patient Records at Mayo Clinic: Lessons Learned From the First 100 Patients in Dr Henry S. Plummer's Dossier Model', *Mayo Clin Proc*, vol. 83, no. 12, pp. 1396–1399, Dec. 2008, doi: 10.4065/83.12.1396.

[3] C. Kainz, R. Lassmann, H. Schaffer, E. Hanzal, and J. Deutinger, 'Survey of computerized obstetric information systems in Austria', *Arch Gynecol Obstet*, vol. 252, no. 2, pp. 87–91, Dec. 1992, doi: 10.1007/BF02389633/METRICS.

[4] W. M. Tierney, M. E. Miller, J. M. Overhage, and C. J. McDonald, 'Physician Inpatient Order Writing on Microcomputer Workstations: Effects on Resource Utilization', *JAMA*, vol. 269, no. 3, pp. 379–383, Jan. 1993, doi: 10.1001/JAMA.1993.03500030077036.

[5] K. T. Kadakia, M. D. Howell, and K. B. Desalvo, 'Modernizing Public Health Data Systems: Lessons From the Health Information Technology for Economic and Clinical Health (HITECH) Act', *JAMA*, vol. 326, no. 5, pp. 385–386, Aug. 2021, doi: 10.1001/JAMA.2021.12000.

[6] J. Metsallik, P. Ross, D. Draheim, and G. Piho, 'Ten Years of the e-Health System in Estonia', 2019.

[7] E. H. W. Kluge, 'Advanced patient records: Some ethical and legal considerations touching medical information space', *Methods Inf Med*, vol. 32, no. 2, pp. 95–103, 1993, doi: 10.1055/S-0038-1634903/ID/BR1634903-8/BIB.

[8] P. B. Jensen, L. J. Jensen, and S. Brunak, 'Mining electronic health records: towards better research applications and clinical care', *Nature Reviews Genetics 2012 13:6*, vol. 13, no. 6, pp. 395–405, May 2012, doi: 10.1038/nrg3208.

[9] 'ICD-10 Version:2019'. Accessed: Mar. 25, 2024. Available: https://icd.who.int/browse10/2019/en

[10] 'World Health Organization (WHO)'. Accessed: Mar. 25, 2024. Available: https://www.who.int/

[11] K. Rahu, E. Palo, and M. Rahu, 'Diminishing Trend in Alcohol Poisoning Mortality in Estonia: Reality or Coding Peculiarity?', *Alcohol and Alcoholism*, vol. 46, no. 4, pp. 485–489, Jul. 2011, doi: 10.1093/ALCALC/AGR046.

[12] L. Manchikanti, F. J. E. Falco, and J. A. Hirsch, 'Ready or not! Here comes ICD-10', doi: 10.1136/neurintsurg-2011.

[13] M. Deschepper, W. Waegeman, D. Vogelaers, and K. Eeckloo, 'Using structured pathology data to predict hospital-wide mortality at admission', *PLoS One*, vol. 15, no. 6, p. e0235117, Jun. 2020, doi: 10.1371/JOURNAL.PONE.0235117.

[14] 'RHK: paringute sooritamine'. Accessed: May 21, 2024. Available: https://rhk.sm.ee

[15] G. Lippi and M. Plebani, 'A modern and pragmatic definition of Laboratory Medicine', *Clin Chem Lab Med*, vol. 58, no. 8, p. 1171, Aug. 2020, doi: 10.1515/CCLM-2020-0114/MACHINEREADABLECITATION/RIS.

[16] Z. E. Khattak, H. El Sharu, and B. S. Bhutta, 'Overview on Ordering and Evaluation of Laboratory Tests', *StatPearls*, Aug. 2023, Accessed: Apr. 21, 2024. Available: https://www.ncbi.nlm.nih.gov/books/NBK570615/

[17] ‘Clinical Biochemistry Handbook Clinical Biochemistry Laboratory User Guide’, 2023.

[18] ‘Definition of reference range - NCI Dictionary of Cancer Terms - NCI’. Accessed: Apr. 21, 2024. Available: https://www.cancer.gov/publications/dictionaries/cancer-terms/def/reference-range#

[19] S. Ferraro, F. Braga, and M. Panteghini, ‘Laboratory medicine in the new healthcare environment’, *Clin Chem Lab Med*, vol. 54, no. 4, pp. 523–533, Apr. 2016, doi: 10.1515/CCLM-2015-0803/MACHINEREADABLECITATION/RIS.

[20] W. J. Marshall and M. Lapsley, ‘Uses of biochemical data in clinical medicine’, *Clinical Biochemistry: Metabolic and Clinical Aspects: Third Edition*, pp. 1–5, Jan. 2014, doi: 10.1016/B978-0-7020-5140-1.00001-8.

[21] P. M. Nadkarni, L. Ohno-Machado, and W. W. Chapman, ‘Natural language processing: An introduction’, *Journal of the American Medical Informatics Association*, vol. 18, no. 5, pp. 544–551, Sep. 2011, doi: 10.1136/AMIAJNL-2011-000464/3/AMIAJNL-2011-000464FIG4.JPEG.

[22] S. Kaji and S. Kida, ‘Overview of image-to-image translation by use of deep neural networks: denoising, super-resolution, modality conversion, and reconstruction in medical imaging’, *Radiol Phys Technol*, vol. 12, no. 3, pp. 235–248, Sep. 2019, doi: 10.1007/S12194-019-00520-Y.

[23] A. S. Lundervold and A. Lundervold, ‘An overview of deep learning in medical imaging focusing on MRI’, *Z Med Phys*, vol. 29, no. 2, pp. 102–127, May 2019, doi: 10.1016/J.ZEMEDI.2018.11.002.

[24] Y. Wang *et al.*, ‘A clinical text classification paradigm using weak supervision and deep representation’, *BMC Med Inform Decis Mak*, vol. 19, no. 1, Jan. 2019, doi: 10.1186/S12911-018-0723-6.

[25] L. Du, C. Xia, Z. Deng, G. Lu, S. Xia, and J. Ma, ‘A machine learning based approach to identify protected health information in Chinese clinical text’, *Int J Med Inform*, vol. 116, pp. 24–32, Aug. 2018, doi: 10.1016/J.IJMEDINF.2018.05.010.

[26] I. Giacomelli, S. Jha, R. Kleiman, D. Page, and K. Yoon, ‘Privacy-Preserving Collaborative Prediction using Random Forests’, *AMIA Summits on Translational Science Proceedings*, vol. 2019, p. 248, 2019, Accessed: Mar. 24, 2024. Available: /pmc/articles/PMC6568057/

[27] M. G. Kahn *et al.*, ‘A Harmonized Data Quality Assessment Terminology and Framework for the Secondary Use of Electronic Health Record Data’, *EGEMS (Wash DC)*, vol. 4, no. 1, p. 18, Sep. 2016, doi: 10.13063/2327-9214.1244.

[28] M. Ivanović and Z. Budimac, ‘An overview of ontologies and data resources in medical domains’, *Expert Syst Appl*, vol. 41, no. 11, pp. 5158–5166, Sep. 2014, doi: 10.1016/J.ESWA.2014.02.045.

[29] L. Leitsalu *et al.*, ‘Cohort Profile: Estonian Biobank of the Estonian Genome Center, University of Tartu’, *Int J Epidemiol*, vol. 44, no. 4, pp. 1137–1147, Aug. 2015, doi: 10.1093/IJE/DYT268.

[30] ‘Human Genes Research Act–Riigi Teataja’. Accessed: Apr. 20, 2024. Available: https://www.riigiteataja.ee/en/eli/531102013003/consolide

[31] J. Priisalu and R. Ottis, ‘Personal control of privacy and data: Estonian experience’, *Health Technol (Berl)*, vol. 7, no. 4, pp. 441–451, Dec. 2017, doi: 10.1007/S12553-017-0195-1/METRICS.

[32] B. P. Prins *et al.*, 'Advances in Genomic Discovery and Implications for Personalized Prevention and Medicine: Estonia as Example', *J Pers Med*, vol. 11, no. 5, p. 358, May 2021, doi: 10.3390/JPM11050358.

[33] 'Estonian Biobank'. Accessed: Apr. 08, 2024. Available: https://genomics.ut.ee/en/content/estonian-biobank

[34] E. Dervić *et al.*, 'Unraveling cradle-to-grave disease trajectories from multilayer comorbidity networks', *npj Digital Medicine 2024 7:1*, vol. 7, no. 1, pp. 1–12, Mar. 2024, doi: 10.1038/s41746-024-01015-w.

[35] A. Giannoula, E. Centeno, M. A. Mayer, F. Sanz, and L. I. Furlong, 'A system-level analysis of patient disease trajectories based on clinical, phenotypic and molecular similarities', *Bioinformatics*, vol. 37, no. 10, pp. 1435–1443, Jun. 2021, doi: 10.1093/BIOINFORMATICS/BTAA964.

[36] S. T. Skou *et al.*, 'Multimorbidity', *Nat Rev Dis Primers*, vol. 8, no. 1, p. 48, Jul. 2022, doi: 10.1038/S41572-022-00376-4.

[37] E. Fabbri *et al.*, 'Editor's choice: Aging and the Burden of Multimorbidity: Associations With Inflammatory and Anabolic Hormonal Biomarkers', *J Gerontol A Biol Sci Med Sci*, vol. 70, no. 1, p. 63, Jan. 2015, doi: 10.1093/GERONA/GLU127.

[38] H. C. Fraser *et al.*, 'Biological mechanisms of aging predict age-related disease co-occurrence in patients', *Aging Cell*, vol. 21, no. 4, Apr. 2022, doi: 10.1111/ACEL.13524.

[39] J. J. Miranda *et al.*, 'Understanding the rise of cardiometabolic diseases in low- and middle-income countries', *Nat Med*, vol. 25, no. 11, pp. 1667–1679, Nov. 2019, doi: 10.1038/S41591-019-0644-7.

[40] M. Fortin, L. Lapointe, C. Hudon, A. Vanasse, A. L. Ntetu, and D. Maltais, 'Multimorbidity and quality of life in primary care: A systematic review', *Health Qual Life Outcomes*, vol. 2, no. 1, pp. 1–12, Sep. 2004, doi: 10.1186/1477-7525-2-51/TABLES/2.

[41] T. Lehnert *et al.*, 'Review: health care utilization and costs of elderly persons with multiple chronic conditions', *Med Care Res Rev*, vol. 68, no. 4, pp. 387–420, Aug. 2011, doi: 10.1177/1077558711399580.

[42] B. Guthrie, K. Payne, P. Alderson, M. E. T. McMurdo, and S. W. Mercer, 'Adapting clinical guidelines to take account of multimorbidity', *BMJ*, vol. 345, no. 7878, Oct. 2012, doi: 10.1136/BMJ.E6341.

[43] C. Boyd *et al.*, 'Decision Making for Older Adults With Multiple Chronic Conditions: Executive Summary for the American Geriatrics Society Guiding Principles on the Care of Older Adults With Multimorbidity', *J Am Geriatr Soc*, vol. 67, no. 4, pp. 665–673, Apr. 2019, doi: 10.1111/JGS.15809.

[44] G. Onder *et al.*, 'Italian guidelines on management of persons with multimorbidity and polypharmacy', *Aging Clin Exp Res*, vol. 34, no. 5, pp. 989–996, May 2022, doi: 10.1007/S40520-022-02094-Z.

[45] A. B. Jensen *et al.*, 'Temporal disease trajectories condensed from population-wide registry data covering 6.2 million patients', *Nature Communications 2014 5:1*, vol. 5, no. 1, pp. 1–10, Jun. 2014, doi: 10.1038/ncomms5022.

[46] T. Siggaard *et al.*, 'Disease trajectory browser for exploring temporal, population-wide disease progression patterns in 7.2 million Danish patients', *Nature Communications 2020 11:1*, vol. 11, no. 1, pp. 1–10, Oct. 2020, doi: 10.1038/s41467-020-18682-4.

[47] ‘The Danish Disease Trajectory Browser’. Accessed: Apr. 04, 2024. Available: http://dtb.cpr.ku.dk/

[48] F. Dabek, E. Jimenez, and J. J. Caban, ‘A timeline-based framework for aggregating and summarizing electronic health records’, *2017 IEEE Workshop on Visual Analytics in Healthcare, VAHC 2017*, pp. 55–61, Jun. 2018, doi: 10.1109/VAHC.2017.8387501.

[49] J. S. Hirsch *et al.*, ‘HARVEST, a longitudinal patient record summarizer’, *J Am Med Inform Assoc*, vol. 22, no. 2, p. 263, 2015, doi: 10.1136/AMIAJNL-2014-002945.

[50] A. Faiola and C. Newlon, ‘Advancing critical care in the ICU: A human-centered biomedical data visualization systems’, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6779 LNCS, pp. 119–128, 2011, doi: 10.1007/978-3-642-21716-6_13/COVER.

[51] N. Steinhauer, M. Hörbrügger, A. Braun, T. Tüting, S. Oeltze-Jafra, and J. Müller, ‘Comprehensive Visualization of Longitudinal Patient Data for the Dermatological Oncological Tumor Board’, 2020, doi: 10.2312/evs.20201067.

[52] Y. Zhang, K. Chanana, and C. Dunne, ‘IDMVis: Temporal Event Sequence Visualization for Type 1 Diabetes Treatment Decision Support’, *IEEE Trans Vis Comput Graph*, vol. 25, no. 1, pp. 512–522, Jan. 2018, doi: 10.1109/TVCG.2018.2865076.

[53] B. S. Glicksberg *et al.*, ‘PatientExploreR: an extensible application for dynamic visualization of patient clinical history from electronic health records in the OMOP common data model’, *Bioinformatics*, vol. 35, no. 21, pp. 4515–4518, Nov. 2019, doi: 10.1093/BIOINFORMATICS/BTZ409.

[54] ‘Data Standardization – OHDSI’. Accessed: Apr. 10, 2024. Available: https://www.ohdsi.org/data-standardization/

[55] ‘PatientexploreR’. Accessed: Apr. 11, 2024. Available: https://comphealth.ucsf.edu/app/patientexplorer

[56] H. Siirtola, R. Raisamo, M. P. Reeve, J. Gracia-Tabuenca, V. Llorens, and S. S. Padmanabhuni, ‘LifeTrack: Decades of EHR Data in a Single View’, *Proceedings of the International Conference on Information Visualisation*, pp. 390–395, 2023, doi: 10.1109/IV60283.2023.00072.

[57] C. D. G. Linhares *et al.*, ‘ClinicalPath: A Visualization Tool to Improve the Evaluation of Electronic Health Records in Clinical Decision-Making’, *IEEE Trans Vis Comput Graph*, vol. 29, no. 10, pp. 4031–4046, Oct. 2023, doi: 10.1109/TVCG.2022.3175626.

[58] ‘FinnGen: an expedition into genomics and medicine | FinnGen’. Accessed: Apr. 11, 2024. Available: https://www.finngen.fi/en

[59] ‘Dash Enterprise: The Premier Data App Platform for Python’. Accessed: Apr. 23, 2024. Available: https://plotly.com/dash/

[60] ‘NumPy -’. Accessed: Apr. 27, 2024. Available: https://numpy.org/

[61] ‘kaleido · PyPI’. Accessed: Apr. 24, 2024. Available: https://pypi.org/project/kaleido/

[62] ‘Dash Bootstrap Components’. Accessed: Apr. 27, 2024. Available: https://dash-bootstrap-components.opensource.faculty.ai/

[63] ‘pandas documentation — pandas 2.2.2 documentation’. Accessed: Apr. 27, 2024. Available: https://pandas.pydata.org/docs/index.html

[64] ‘utils — dateutil 2.8.2 documentation’. Accessed: Apr. 27, 2024. Available: https://dateutil.readthedocs.io/en/2.8.2/utils.html

[65] ‘webbrowser — Convenient web-browser controller — Python 3.12.3 documentation’. Accessed: May 04, 2024. Available: https://docs.python.org/3/library/webbrowser.html

[66] ‘Visual Studio Code - Code Editing. Redefined’. Accessed: Apr. 24, 2024. Available: https://code.visualstudio.com/

[67] ‘Estonian Biobank’. Accessed: Apr. 27, 2024. Available: https://genomics.ut.ee/en/content/estonian-biobank

APPENDIX

1. GitHub repository with the code and user guide

https://github.com/PlanetWyh/Medi_Vizz

2. User Opinion by Anu Reigo, MD (University of Tartu)

Having tested the Medi_Vizz application as a first-time user, I share my experience as outlined below.

Following the shared link to Medi_Vizz and the materials therein, my onboarding to the app was generally fluent. First, I read the instructions in the provided manual (Medi_Vizz_User_Guide_Windows.pdf), and it was understandable how to proceed at every step.

The only setback for me was before the Python libraries installing part, where it seemed that my computer had resisted the Python version renewal from 3.11.9 to 3.12.0, so I turned back and removed the older version and then replaced it with the new one.

I think installing the libraries is guided in a sufficient and appropriate way for those already somewhat familiar with Python; otherwise some closer assistance may be needed (and probably can be provided by the author, as indicated in the very beginning of the manual, or on-site by any more experienced user).

Checking the data preprocessing requirements and examples (ICD-10-example-file.txt; LAB-MES-example-file.txt) is a prerequisite that reminds the user how important it really is to build up databases (medical, scientific – or actually, any) in a well-structured and unified manner, and with built-in quality control and data normalization functions. Otherwise, it remains to the user to do it after raw data download (usually in *.csv, *.xlsx, etc) and will take a lot of time. However, even then it would be worth the effort, as the Medi_Vizz app will do the work in the final steps where understanding the patient situation comes much more easily than scrolling through data tables (conventional „scientific“ view) or clicking between different files and interfaces (conventinal „medical“ view).

The Medi_Vizz customization options and functionalities are convenient to get an idea of an individual before first face-to-face appointment in a biobank (scientific) setting or to understand a patient’s general picture before giving an opinion or planning further steps in diagnostics or treatment in real-life practical (medical) settings. As mentioned by the author, this task is yet a

real challenge today, especially when thinking of the widening scope and limited resources within the expanding field of personalised medicine.

Within the task setup and time frame for this software development, I consider the Medi_Wizz application really functional and would be eager to try it in a future feedback project in EstBB. With its logical structure and independent on-site working capability, Medi_Wizz has potential for further amendments without adding much computing load. Most importantly, it can save time and other resources of the users while supporting them to build quality and comprehensibility.

NON-EXCLUSIVE LICENCE TO REPRODUCE THESIS AND MAKE THESIS PUBLIC

I, **Uscinnia Dyn'ko**,

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright, my thesis **Medi_Vizz: A tool for concurrent visualisation of medical data**, supervised by **Associate Professor Toomas Haller and MSc Miriam Nurm.**

2. I grant the University of Tartu a permit to make the work specified in p. 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 3.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright.

3. I am aware of the fact that the author retains the rights specified in p. 1 and 2.

4. I certify that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Uscinnia Dyn'ko

22/05/2024