A Web Platform for Analysis of Multivariate Heterogeneous Biomedical Time-Series - a Preliminary Report

Alan Jovic [1], Davor Kukolja [1], Kresimir Jozic [2], Marko Horvat [3]
[1] University of Zagreb Faculty of Electrical Engineering and Computing, Unska 3, HR-10000 Zagreb, Croatia
[2] INA - industrija nafte, d.d., Avenija Veceslava Holjevca 10, p.p. 555, HR-10002 Zagreb, Croatia
[3] Zagreb University of Applied Sciences, Konavoska 2, HR-10110, Zagreb, Croatia
alan.jovic@fer.hr

Abstract - **Biomedical time-series analysis is a diverse field that includes biomedical engineering, computer science, and medical achievements, with the goal to save human lives and improve the quality of healthcare. In this paper, we present a preliminary report on construction of an innovative web platform for heterogeneous multivariate biomedical time-series analysis. The platform will feature data upload, preprocessing, feature extraction, model construction, visualization of signals and disorders, and reporting. Several scenarios of use will be prepared for a user, depending on his research or practice goals. An expert system for feature recommendation, specifically designed to support clinical decisions, will be implemented in the platform. In the current phase of research on our project, we describe the features of the platform, including some of its technological details, and the way in which we designed scenarios of use. Future papers will focus on specific aspects of the platform.**

Keywords **- Biomedical time-series analysis; Machine learning; Web platform; Expert system; Visualization; Use case scenario**

I. INTRODUCTION

In recent years, the field of web-based telemedicine, especially in the context of increasingly resource-constrained healthcare system, has been rapidly evolving. Many reasons can be given for explaining this trend, but, according to [1], the two most important ones are 1) the need to reduce the cost of healthcare expenditure in developed countries, and 2) to facilitate access to a better healthcare. However, despite the advantages of telemedicine for remote diagnostic purposes, an expert to analyze the biomedical data is still needed.

Hence, a step further in medicine would be the development of an efficient and upgradeable system for automatic classification of human body disorders based on the analysis of multiple heterogeneous biomedical signals (heart rate variability (HRV), ECG, EEG, etc.), which would help medical specialists in diagnostics and early detection of various diseases. Therefore, this paper presents a conceptual overview of a web platform that would support multivariate analysis of heterogeneous biomedical time-series (BTS). The challenges in design and construction of such a platform are great, because the platform is expected to support web browser data input, the analysis of a large number of different BTS and their individual, domain specific features, their visualization, and detection, classification, or prediction of various health disorders based on machine learning algorithms.

Finding significant patterns in a time-series is an unsolved problem in science, neither in a general scenario of modeling and comparing multiple time-series of undetermined origin, nor in the specific scenarios within particular domains of application [2]. Nevertheless, a significant improvement in the direction of general time-series modeling was brought by the work of Fulcher et al. [3]. In their work, the authors implemented a large collection of features (over 9 000) and collected numerous time-series (over 35 000) from the majority of scientific areas. One of the major conclusions of the work was that there exists a set of roughly 200 features that characterize sufficiently well the majority of time-series collected from different scientific areas. The features in the study were limited to scalar operations and did not include domain features, except for three categories of HRV measures.

During the implementation of the platform for analysis of BTS, the results of the work made by [3] will be taken in the sense of integration of the general features analyzed in the research with specialized, domain features. The collection of features analyzed in [3] can be enlarged with the set of features implemented in HRVFrame framework [4], which supports a large number of features for domain based and general HRV analysis, with particular attention devoted to nonlinear dynamics features that were shown to improve classification performance of linear feature combinations [5]; or with the set of features implemented in EEGFrame [6] tool that contains a library with a large number of implemented one-dimensional and some multidimensional domain features for EEG analysis.

For the purpose of their ongoing research, Fulcher et al. implemented Comp-Engine toolbox [3]. Its advantage, in the context of biomedical analysis, is the large number of implemented features and the ability of classification of a large number of health disorders. However, it includes the support only for licensed Matlab users. Moreover, specialized domain features are not implemented and large complexity makes it unsuitable for an average user.

This work has been fully supported by the Croatian Science Foundation under the project number UIP-2014-09-6889.

Recently, pervasive and ubiquitous computing systems and their applications in intelligent ambient environment are becoming more and more popular [7]. Nowadays, smartphones and tablets can use wearable physiological sensors to provide medical healthcare from the comfort of one's own home, or during everyday activities. To make this possible, it is necessary to develop web-based systems for transmission of biomedical data collected from users to the server, where they will be analyzed, or examined by an expert. An example of such a solution is given in [7], where a mobile device system for early warning of ECG anomalies was developed. Also, a novel open-source Java-based Android application was presented in [1], offering advanced ECG processing, including signal quality analysis and Atrial Fibrillation screening.

According to Calabrese and Cannataro [8], different works also reported the successful application of cloud computing in healthcare and biomedicine. For example, a cloud-based system for clients with mobile devices or web browsers was developed [9]. The server, upon receiving the data, controls the quality of the received ECG signals and, if necessary, applies appropriate enhancement algorithms. Subsequently, the system extracts HRV features and visualizes these data to the cardiologist. According to their value, he/she can make a diagnosis that is sent to the patient. Although web-based systems for BTS analysis already exist, the vast majority of solutions are currently limited to the analysis of ECG signals. Furthermore, systems which support the classification algorithms can detect only a limited set of health disorders.

II. METHODOLOGY

A. *Scenarios for Platform Use*

In order to accommodate for different research directions and goals in the field of BTS analysis, the platform will support numerous possible scenarios of use. We have divided the analysis process into 8 steps, some of which may be skipped, depending on the user: 1) analysis type selection, 2) scenario selection, 3) input data selection, 4) records inspection, 5) records preprocessing, 6) feature extraction, 7) model construction, and 8) reporting.

For each step of the process, a UML Use Case Diagram (UML 1.4+) was drawn using Astah Community tool [10], depicting user's activities related to this step. The goal was to formalize the requirements for the platform in order to be more efficient and unambiguous when it comes to implementation. The strategy to use UML Use Case Diagrams for such a task is well-known and often used in software engineering in many areas, even where other UML diagram types are not used [11]. An example UML Use Case Diagram for the first step: "Analysis type selection" is shown in Fig. 1.

Basically, this use case diagram shows that we will allow the user to select a goal type for the analysis (e.g. classification) and to select the type of input data records for the platform (e.g. multivariate homogeneous BTS that is EEG). A user could also have selected the analysis of heterogeneous time-series, e.g. EEG and heart beat series. The second step, scenario selection, which we do not depict here because of space limitations, allows further specification of the analysis.

Each use case, e.g. "AT2: Select data type", has a designation (AT2) which links to the requirements document section in which one can describe the use case in more details. Thus, a reasonably complete set of requirements may be provided textually in more detail and visualized using an UML tool. This procedure greatly simplifies further construction of software architecture and, in particular, it drives the construction of graphical user interface in a browser, which is the only thing a user sees in the end.

The UML Use Case Diagrams may be used in addition to other methods in requirements specification for depicting the scenarios of software use, such as flowcharts, UML Activity Diagrams, or other diagrams. For instance, in Fig. 2, we show an example of a flow-based diagram that depicts a complete scenario.

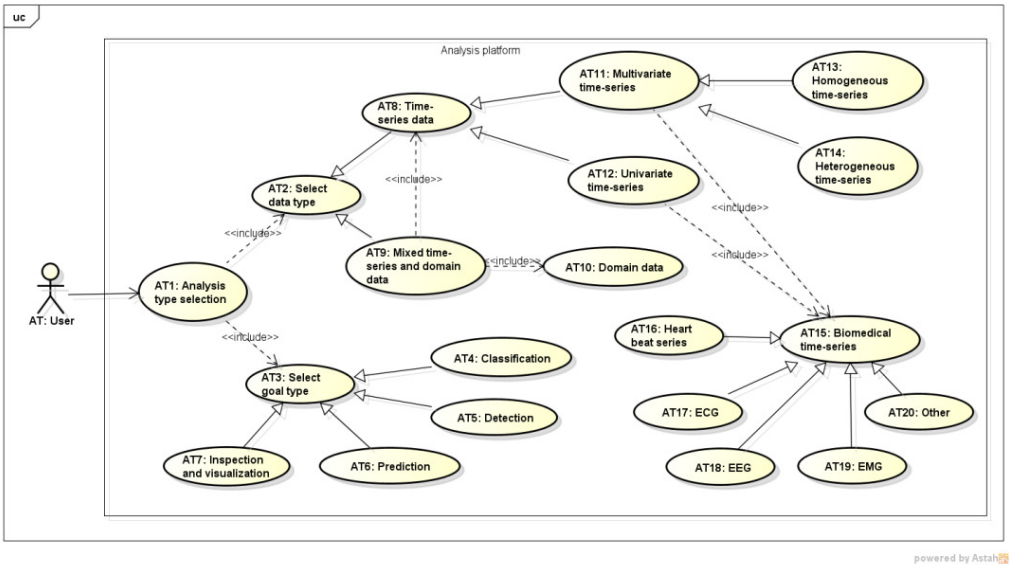

Figure 1. UML Use Case Diagram for the first step of the analysis that will be provided by the platform: "Analysis type selection"

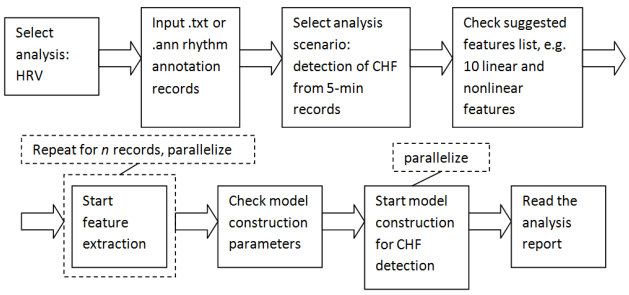

Figure 2. An example flow-based diagram of analysis scenario: model construction for detection of congestive heart failure (CHF) based on heart rate variability (HRV) from 5-minute segments

B. *Data input*

The platform will enable multiple heterogeneous BTS analysis. Therefore, there is a need for supporting input data records containing a variable number of data arrays, mostly recorded from leads on a patient. We have considered a number of formats, and in the end, we opted to support: 1) European data format (EDF/EDF+), 2) textual format for signals and annotations, and 3) image formats JPEG and TIFF.

The main reasons to support formats 1) and 2) are their widespread use in the biomedical signal analysis community and their openness. EDF/EDF+ is a well-known standard format for polygraphic recordings supported by the majority of recording devices and available from open-access internet databases. It is mostly used in EEG and polysomnograph recording, but it can also accommodate ECG, EMG and other information sources [12]. Textual format files supported by the platform will have the same format as provided by the PhysioBank ATM Toolbox tools *rdsamp* and *rdann*. Lastly, it is conceived that the platform will allow uploading images of the record sections (in .tiff or .jpeg files) in order to enable automatic image processing and analysis. This will be supported only for a limited subset of signal sources and tasks. For example, a digitized image of a standard single-lead ECG or 12-lead ECG may be uploaded in order to analyze it for possible disorders. However, as various 12-lead ECG systems exist in practice, we will have to opt for one or two of these formats. The exact format and constraints for such images is the subject of future work. The platform will allow batch input of (possibly) many patient data records in order to allow for reliable model construction.

C. *Preprocessing, visualization and feature extraction*

The visualization part of the platform will perform 2D visualization of the signals from uploaded records, including: segments selection, temporal and amplitude scaling, lead(s) selection, and header information inspection. For preprocessing, the plan is to implement baseline correction, noise and other filtering techniques, as well as different windowing procedures. The user would be able to select which record segments will be used for feature extraction. After filtering and before feature extraction, several data transformations will be possible to apply, such as time domain (e.g. PCA), frequency domain (e.g. FFT), and time-frequency domain (e.g. WT) transformations. Thereafter, 3D visualization of patient disorders or feature extraction will proceed, depending on the desired analysis goal. A separate module for 3D visualization of medical conditions from medical records is envisioned, which will employ WebGL and which will be similar to volume rendering web tools such as arivis WebView [13].

For BTS feature extraction, we plan to implement both domain-specific features (e.g. RMSSD for HRV) as well as general time-series features (e.g. approximate entropy, correlation dimension, mutual dimension, etc.). The algorithms from HRVFrame [4], EEGFrame [6], and Comp-Engine [3] will be considered, implemented or re-factored, and verified. Furthermore, the plan of the project is to develop additional domain specific feature extraction frameworks, for ECG and EMG at least, and, if deemed feasible, for multiple blood pressure types, breathing impedance, CO_2 saturation, gait dynamics, and galvanic skin resistance. All of the domain specific frameworks will be developed based on scientific work published in medical and biomedical engineering journals as well as medical guidelines, and through consultations with at least two medical experts working on the project.

An important part of the platform is an expert system that will, based on the available information about the chosen input biomedical signals, goal of the analysis and selected scenario, recommend a list of features and features' parameters for extraction, with the possibility of user's intervention in the selection. Hence, although the expert system will provide an opportunity for automatic feature extraction, its intention is not to replace the medical experts, but rather to support their decisions, based on the available medical knowledge.

Feature extraction will be parallelized so that the platform optimally utilizes the available system resources, including multiple CPU cores and general purpose GPU (GPGPU) parallelization through the use of JCuda or jocl libraries.

D. *Machine learning algorithms and reporting*

Dimensionality reduction methods will be offered to the user prior to model construction in the case where a large number of features were extracted beforehand, either because the expert system recommended it, or because a user specified it. Typical filters and wrappers will be made possible to select, according to the most recent relevant literature, depending on the goal [14]. The idea will be to remove any irrelevant and redundant features from the dataset in order to speed up the model construction procedure and possibly improve model accuracy.

Several machine learning algorithms, with their corresponding hyperparameters will be offered to the user after the feature extraction part is successfully completed. For detection and classification models, tree-based and SVM-based algorithms will be provided (e.g. C4.5, random forest, *C*-SVM or SMO), as these methods are fast and accurate, while some of them (e.g. C4.5) retain comprehensiveness of the obtained classification rules [15]. It will be possible to evaluate the data using standard evaluation procedures (i.e. holdout, cross-validation), both patient-wise (personalized) or regardless of the patient [16].

For reporting purposes, the platform will provide representation of the model results using Java-based

JasperReports Library in a web form, with the possibility to export to PDF, Excel, OpenOffice, and Word documents [17].

E. Platform architecture

It was decided to create the analysis platform as a web application. In that manner, end user base will be larger, whilst application development and maintenance will be less demanding [18].

Java was selected for server side mainly because of a large base of existing libraries for signal processing, data parsing, machine learning, and parallelization. Because of the fact that the platform's user interface will be displayed in a web browser, Java application server with all its features is not needed. Only a few components, such as RESTful server, the user authentication library, and object-relational mapping library (e.g. Hibernate) for database access are needed, thus lowering memory requirements on the server.

Two approaches are possible: 1) Java application server with only a minimal set of features, or 2) standalone framework, which provides the necessary functionality (e.g. Spring Boot, which bootstraps the Spring application context inside of an embedded servlet container [19]). In the first case, a Java application server needs to be installed on one or more servers and administered during application usage (complexity of administration rises with the number of added servers). In the second case, Java application just needs to be copied to one or more servers and then started, without any administration. Both cases will be tested with criteria such as performance, memory consumption, and ease of administration.

On the client side, HTML5, Typescript, and CSS3 will be used for the design of web pages. HTML5 will be used, because it provides the technologies needed for platform development, such as WebGL and Web components. Typescript is a strict superset of JavaScript, which compiles to the selected version of JavaScript (ES3, ES5, ES6) [20]. It corrects some of JavaScript deficiencies, of which the most significant one is the lack of static typing. The framework chosen for the client-side platform development is Angular 2. Angular's way of displaying and manipulating with data is through the use of components [21]. It also supports routing, dependency injection, and has a cross-platform support for desktop and mobile web applications. Finally, visual design of web pages will be enhanced using Bootstrap CSS3 styles.

III. CONCLUSION

A thorough examination of contemporary technologies revealed a possibility to construct a useful web platform for both researchers and medical personnel in the field of multivariate heterogeneous BTS analysis. The platform will feature a complete process of BTS data records analysis and visualization, with special attention devoted to efficiency, ease-of-use, and strong medical knowledge support, made available through implementation of an innovative expert recommendation system. The described methodology should benefit researchers with similar biomedical platforms in mind.

In the future, we plan to report on various aspects of the platform in more detail, as the work on the platform progresses.

REFERENCES

[1] J. Oster, J. Behar, R. Colloca, Q. Li, Q. Li, and G. D. Clifford, "Open source Java-based ECG analysis software and Android app for atrial fibrillation screening," in Proceedings of the Computing in Cardiology Conference (CinC 2013), IEEE, 2013, pp. 731-734.

[2] M. Kim, "Probabilistic Sequence Translation-Alignment Model for Time-Series Classification," IEEE Trans. Knowl. Data Eng., vol. 26, pp. 426-437, February 2014.

[3] B. D. Fulcher, M. A. Little, and N. S. Jones, "Highly comparative time-series analysis: the empirical structure of time series and their methods," J. R. Soc. Interface, vol. 10, p. 20130048, April 2013.

[4] A. Jovic, N. Bogunovic, and M. Cupic, "Extension and Detailed Overview of the HRVFrame Framework for Heart Rate Variability Analysis," in Proceedings of the Eurocon 2013 Conference, I. Kuzle, T. Capuder, and H. Pandzic, Eds. Zagreb: IEEE Press, 2013, pp. 1757-1763.

[5] A. Jovic and N. Bogunovic, "Evaluating and Comparing Performance of Feature Combinations of Heart Rate Variability Measures for Cardiac Rhythm Classification," Biomed. Signal Process. Control, vol. 7, pp. 245-255, May 2012.

[6] A. Jovic, L. Suc, and N. Bogunovic, "Feature extraction from electroencephalographic records using EEGFrame framework," in Proceedings of the 36th International Convention MIPRO 2013, P. Biljanović, Ed. Rijeka: MIPRO Croatian Society, 2013, pp. 1237-1242.

[7] A. Szczepanski and K. Saeed, "A Mobile Device System for Early Warning of ECG Anomalies," Sensors, vol. 14, pp. 11031-11044, 2014.

[8] B. Calabrese and M. Cannataro, "Cloud Computing in Healthcare and Bioinformatics," Scalable Computing: Practice and Experience, vol. 16, pp. 1-18, 2015.

[9] H. Xia, I. Asif, and X. Zhao, "Cloud-ECG for real time ECG monitoring and analysis," Comput. Meth. Programs Biomed., vol. 110, pp. 253-259, June 2013.

[10] Change Vision Inc., "Astah, Community Edition," http://astah.net/editions/community, last accessed on: 2016-02-03.

[11] G. Reggio, M. Leotta, and F. Ricca, "Who Knows/Uses What of the UML: A Personal Opinion Survey," Lecture Notes in Computer Science, vol. 8767, J. Dingel, W. Schulte, I. Ramos, S. Abrahao, and E. Insfran, Eds. Springer International Publishing, 2014, pp. 149-165.

[12] B. Kemp and J. Olivan, "European data format 'plus' (EDF+), an EDF alike standard format for the exchange of physiological data," Clin. Neurophysiol., vol. 114, pp. 1755-1761, September 2003.

[13] Arivis Vision, "Arivis WebView," http://vision.arivis.com/en/arivis-WebView, last accessed on: 2016-02-03.

[14] J. Tang, S. Alelyani, and H. Liu, "Feature Selection for Classification: A Review," in Data Classification: Algorithms and Applications, C. C. Aggarwal, Ed. Boca Raton: CRC Press, 2014, pp. 37-64.

[15] D. Kotsakos and D. Gunopulos, "Time Series Data Classification," in Data Classification: Algorithms and Applications, C. C. Aggarwal, Ed. Boca Raton: CRC Press, 2014, pp. 365-378.

[16] J. Park and K. Kang, "PcHD: personalized classification of heartbeat types using a decision tree," Comput. Biol. Med., vol. 54, pp. 79-88, November 2014.

[17] TIBCO Software, "JasperReportsLibrary," http://community.jaspersoft.com/project/jasperreports-library, last accessed on: 2016-02-04.

[18] J. Harjono, G. Ng, D. Kong, and J. Lo, "Building smarter web applications with HTML5," in Proceedings of the 2010 Conference of the Center for Advanced Studies on Collaborative Research, H. A. Müller, A. Ryman, A. W. Kark, Eds. Riverton: IBM Corp., 2010, pp. 402-403.

[19] Pivotal Software, "Spring Boot," http://projects.spring.io/spring-boot, last accessed on: 2016-02-04.

[20] Microsoft Corporation, "TypeScript," http://www.typescriptlang.org, last accessed on: 2016-02-04

[21] Google Inc., "Angular 2," https://angular.io, last accessed on: 2016-02-04.