An Integrative Clinical Database and Diagnostics Platform for Biomarker Identification and Analysis in Ion Mobility Spectra of Human Exhaled Air

Till Schneider [1,3,*], Anne-Christin Hauschild [1,2,*,], Jörg Ingo Baumbach [5] and Jan Baumbach [1,2,4]**

[1]Computational Systems Biology Group, Max Planck Institute for Informatics, D-66123, Saarbrücken, Germany, http://www.mpi-inf.mpg.de/

[2]Cluster of Excellence for Multimodal Computing and Interaction. Saarland University, D-66123 Saarbrücken, Germany http://www.mmci.uni-saarland.de/

[3]KIST Europe, Department Microfluidics and Clinical Diagnostics, D-66123, Saarbrücken, Germany, http://www.kist-europe.de/

[4]Computational Biology Group, Department of Mathematics and Computer Science, University of Southern Denmark, DK-5230, Odense, Denmark, http://www.sdu.dk/en/

[5]B & S Analytik, BioMedizinZentrum Dortmund, D-44227, Dortmund, Germany, http://www.bs-analytik.de

Summary

Over the last decade the evaluation of odors and vapors in human breath has gained more and more attention, particularly in the diagnostics of pulmonary diseases. Ion mobility spectrometry coupled with multi-capillary columns (MCC/IMS), is a well known technology for detecting volatile organic compounds (VOCs) in air. It is a comparatively inexpensive, non-invasive, high-throughput method, which is able to handle the moisture that comes with human exhaled air, and allows for characterizing of VOCs in very low concentrations. To identify discriminating compounds as biomarkers, it is necessary to have a clear understanding of the detailed composition of human breath. Therefore, in addition to the clinical studies, there is a need for a flexible and comprehensive centralized data repository, which is capable of gathering all kinds of related information. Moreover, there is a demand for automated data integration and semi-automated data analysis, in particular with regard to the rapid data accumulation, emerging from the high-throughput nature of the MCC/IMS technology. Here, we present a comprehensive database application and analysis platform, which combines metabolic maps with heterogeneous biomedical data in a well-structured manner. The design of the database is based on a hybrid of the entity-attribute-value (EAV) model and the EAV-CR, which incorporates the concepts of classes and relationships. Additionally it offers an intuitive user interface that provides easy and quick access to the platform's functionality: automated data integration and integrity validation, versioning and roll-back strategy, data retrieval as well as semi-automatic data mining and machine learning capabilities. The platform will support MCC/IMS-based biomarker identification and validation. The software, schemata, data sets and further information is publicly available at http://imsdb.mpi-inf.mpg.de.

*The authors wish to be considered as joint first-authors.

**To whom correspondence should be addressed. Email: a.hauschild@mpi-inf.mpg.de

1 Introduction

The achievements in the development of modern analytical techniques especially chromatography and spectrometry, facilitates the replacement of human and animal senses with refined chemical measurements [1].
The analysis of exhaled breath utilizing these techniques is gaining more and more attention, especially in the diagnostics of pulmonary diseases [2], as well as other diseases for instance sarcoidosis [3] or bacterial infections and the monitoring of pharmaceuticals or experiments on bacteria [4]. A major goal of such studies is to determine volatile substances that are relevant to particular objectives. To distinguish between specific substances, proposed as biomarkers, and compounds originating from confounding factors, such as nutrition or live-style, it is necessary to have a clear understanding of the detailed composition of human exhaled air.
Several analytical detection methods for human breath investigations can provide early and fast diagnosis or therapy monitoring. Examples for the most common spectrometric methods currently employed include gas chromatography-mass spectrometry (GC/MS), electronic noses, and ion mobility spectrometry (IMS). We refer the reader to [4] for a detailed summary of these technologies. In the next Section we will introduce the MCC/IMS technique in more detail. It is a comparably inexpensive, high-throughput method, which allows for the characterization of volatile compounds, especially in very low concentrations as required for exhaled air analysis.

1.1 MCC/IMS

The first IMS instruments, developed in the early 1970, were constructed for military applications to detect drugs or explosives. In the last decades, the combination of the IMS and the multi capillary column has evolved to a powerful technology for detecting volatile compounds, which opened up the horizon to many other applications. Due to the growing importance of metabolomics in clinical diagnostics the focus changed towards medical applications [4].
The MCC/IMS technology has several advantages, listed in the following. The device can handle the moisture that comes with exhaled air. It is very sensitive compared to other techniques such as GC/MS (detection limit at nanogram to picogram per liter) and easy to use in every day practice. Taking a measurement is non-invasive and fast (around 5 min).
Within the MCC/IMS the compounds are separated by polarity and mobility as follows: First, entering the instrument, the volatile analytes are driven by a carrier gas and pre-separated inside the MCC. Second, the analytes reach the ionization chamber, where a radioactive ionization source (^{63}Ni) ionizes the carrier gas molecules to provide reactant ions. The target substances are chemically ionized when colliding with these reactant ions and the resulting product ions enter the drift region during the ion shutter opening times. The conditions in the drift-tube (applied electric field; drift gas flow towards ions) guarantee that only ions arrive at the Faraday-Plate. In an ideal case, all molecules are totally separated when they reach the Faraday-Plate and an ion mobility spectrum is generated. Finally, the accumulation of all IMS spectra generates a three-dimensional diagram, called IMS chromatogram. In the special case of human breath analysis, the SpiroScout can be installed upstream of the MCC/IMS, to collect the air from specific parts of the respiratory system [5]. For further details of the IMS technique, see Baumbach *et al.* (2009) [5].

1.2 Pre-processing and Emerging Requirements

An IMS chromatogram has three dimensions: The retention time (RT, time within the MCC), the inverse mobility ($1/K_0$, corresponding to the flight time in the IMS) and finally the electric current (h) measured by the Faraday detector. Given a standardized setting of the MCC/IMS (including the same temperature, carrier gas flow, drift gas flow, electric field, polarity of the MCC), a compound within a gaseous sample occurs as a peak at a specific position of the chromatogram. Small variations in these parameters lead to noise within the measurement. Each chromatogram shows a characteristic signal structure known as the reactant ion peak (RIP) and a signal descent at the right side, which is also considered as a source of perturbations. To detect those peaks corresponding to analytes rather than noise or the RIP, a set of computational methods has been successfully developed in the last ten years. This set comprises tools for RIP-detailing, de-noising, smoothing, peak finding and merging peak sets. These techniques are inter alia described in [4] and the references therein.
Furthermore, this review points out several remaining challenges. Most of all the development of a flexible and comprehensive centralized data repository, which is capable of gathering all kinds of relevant data, in particular potential confounding factors. This will allow for investigating their influence to IMS-based biomarker identification. Therefore, this repository has to include the following information:

- The IMS measurement, including experimental conditions to account for the influence of the experimental setting.
- The result of the pre-processed measurement (list of peaks) and the information on the utilized set of tools to evaluate the optimal combination of pre-processing methods.
- The object-specific attributes, including object attributes (gender and age, for human, or antibiotics resistances, for bacteria, for instance) as well as environmental influences (e.g. nutrition, medication or diseases).

On the one hand the incorporation of this information enables the assessment of the quality of the various pre-processing techniques as well as the experimental settings. On the other hand it allows for the discovery of their influence to IMS chromatogram processing and hence a more accurate analysis. Moreover, there is a demand for automated data integration and semi-automated data analysis, in particular with regard to the rapid data accumulation emerging from the high-throughput nature of the MCC/IMS technology.

1.3 Related Work

In 2007, Baumbach *et al.* [6] developed the first automated systems for IMS data analysis. In the same year, Lesniak developed the first database schema to organize IMS data [7]. Nevertheless, the approaches in that studies are not capable of fulfilling the requirements mentioned above. The database from Lesniak is based on fixed input data, not able to store arbitrary entities, attributes and values, as well as relations between entities, which is essential to make the database adaptable to any kind of biomedical data. The data analysis system from Baumbach *et al.* is also unable to include confounding factors, such as patient data.

Beyond the field of IMS research, there are two areas overlapping with the proposed database framework. The first field covers the advanced processing and analysis of metabolomics and proteomics data, including software tools as MeltDB [8, 9] or the free software library OpenMS [10]. The second field deals with the problem of adequate and flexible data storage. There are several commercial (for instance Oracle Clinical (Oracle Corporation)) and non-commercial (e.g. TrialDB [11] and SenseLab [12]) projects focussing on the development of flexible clinical data repositories. The projects in those fields either do not include heterogeneous biomedical data, or are not able to handle metabolomic data and do not include data validation. For the IMSDB database framework, we aimed to combine the strengths of both research fields, leading to a powerful software solution to handle metabolic data combined with heterogeneous biomedical data.

2 Methods

In this section we first describe the content and format of the data integrated in the IMSDB framework. Furthermore, we present the structure of the data model and give some details on the implementation.

2.1 Data

The IMS measurements, the detected peaks and the heterogeneous object data, are stored in a set of well-defined file types, which are generated by firmware and pre-processing tools described in the previous section. The most important file types are explained in the following.
An **IMS measurement file** comprises experimental and technical information as well as the raw spectra measured by an MCC/IMS instrument. The file format is described in [13]. The *sample ID* variable serves as identifier for sample objects. In the following, we do not distinguish between IMS measurement file and IMS measurement, provided that the meaning can be deduced from the context.

A **PeakAn file** contains information about signals in a particular peak region retrieved from a set of IMS measurements. A peak region can be represented as a rectangle which is defined by two diagonally opposite corner points. These points are defined by the two coordinates $1/K_0$ and RT. In the ideal case, a particular volatile analyte can be related to all detected peaks within a peak region. For each measurement, the maximum signal intensity with related retention time and inverse mobility coordinate in the peak region are stored in the corresponding PeakAn file.

An **object-attribute-table** is a human generated spreadsheet. It provides heterogeneous anonymous attributes of objects, related to examinations at a specific time point. Each examination of an object is referred to as an object case denoted by a row in the table. The columns comprise distinct attribute names and the datatype of the attribute values. In order to identify an object case, an *ID* attribute corresponding to the *sample ID* in the IMS measurement is included in the first column. The name of the corresponding IMS measurement is given in column *imsfile*. A *class* attribute denotes the hospital or institution which the object/patient belongs to. An example for an object-attribute-table is given in Table 1.

Table 1: This table illustrates the structure of the object-attribute-table. The first two rows comprise identifiers and attribute names with corresponding data types in the second row.

id	class	imsfile	attribute1	attribute2	...
string	string	string	boolean	date	...
304856	hospital_patient	file1_ims.csv	yes	01.01.2011	...
305082	hospital_patient	file2_ims.csv	no	01.01.2012	...
...	...	...	...	...	...

2.2 Structured Data Model

The developed data model can be divided into three main components, namely the IMS measurement, generalized data structure and the peak component, which are described in detail in the following. The connected components model the relations between IMS measurements, peaks and objects cases as shown in Figure 1.

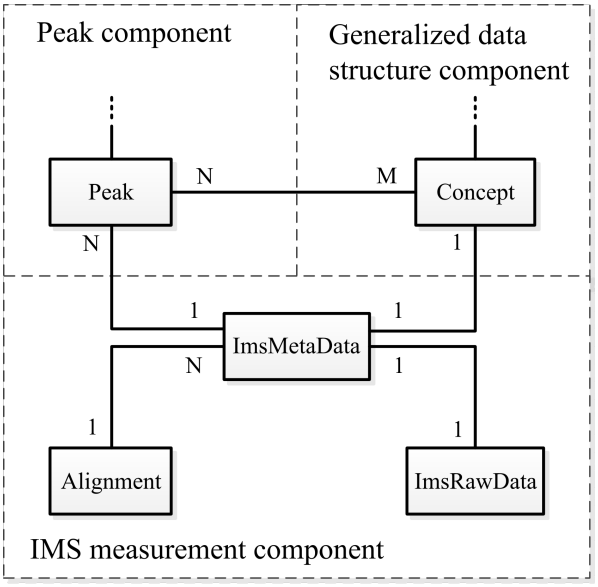

Figure 1: Overview of the three main database components and their relations. The central core entities of each component are linked directly. Auxiliary entities of generalized data structure and peak component can be found in the next paragraph. Associations and maximum cardinalities (one or many) are represented as solid lines labeled with characters 1, N, M.

1) IMS measurement component contains entities related to IMS measurements: *ImsMetaData*, *ImsRawData*, and *Alignment*, see Figure 1. The first represents experimental conditions including all parameters of an IMS measurement file. *ImsRawData* comprises the IMS measurement file. *Alignment* refers to parameters of a linear transformation for each chromatogram.

2) Generalized data structure component, also referred to as ontology component, models objects with arbitrary attributes. In addition, the conception of relations allows for relationships between any two objects, see [14, 15] for related work. The ontology component is similar to an entity-attribute-value (EAV) model for the organization of heterogeneous data, among others, described in [12]. Its main function in this work is to model sparse object data

retrieved from an object-attribute-table. Figure 2 shows the participating entities and relationships.

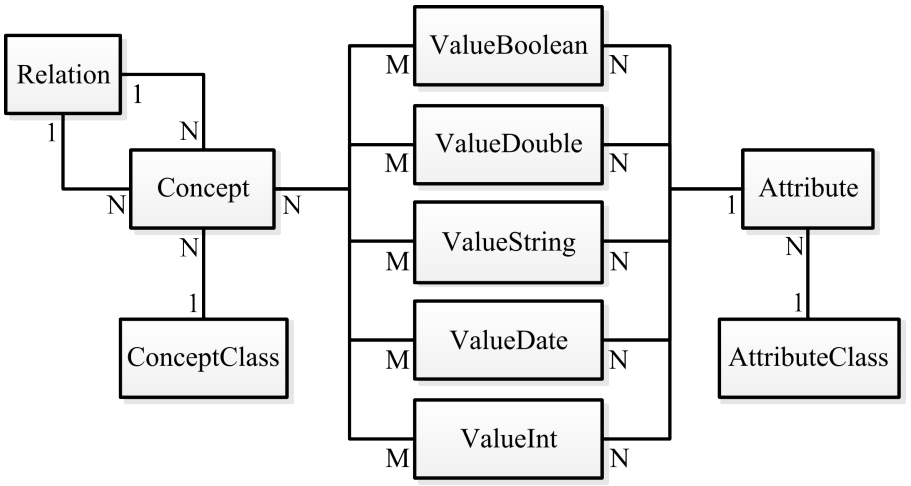

Figure 2: Overview of the generalized data structure (ontology) component. The central entity of this component *Concept* is associated to combinations of *Attribute* and value entities. In addition, a *Relation* is used to associate *Concept* entities to each other.

The central *Concept* entity represents an object, for instance a patient case referred to a row of the object-attribute-table (Table 1). Many-to-many associations between *Concept* and the value entities (*ValueBoolean*, *ValueDate*, *ValueDouble*, *ValueInt*, *ValueString*) represent the assignment of attribute values to the *Concept*. Each of these value entities, which include a value column of the corresponding type, is assigned to an *Attribute* entity by using a many-to-one association. An *Attribute* contains a unique *name* column. The application-logic ensures the attribute-value-pairs to be unique. To organize *Attribute* and *Concept* according to various meta information (e.g. different institutions), they are associated to *AttributeClass* and *ConceptClass* respectively. Finally *Concepts* can be linked to each other by the entity *Relation*. The representation of time series, including the patient history, is thereby modelled with the help of connections between the first occurring case of an object and all following occurrences. The relationship between a sample object (e.g. a patient record) and a corresponding IMS measurement is represented by a one-to-one association between the entities *ImsMetaData* and *Concept* (shown in Figure 1).

3) Peak component offers a semi-generic structure for peak data persistence. This comprises the required peak parameters peak position and peak region and additionally enables arbitrary parameters for instance the maximum peak intensity. Details of the peak component are shown in Supplementary[1] Figure 1. An association between *Peak* and *Concept*, as shown in Figure 1, allows the data model to support any kind of annotation.

2.3 Implementation

The IMS database developed in this work relies on a PostgreSQL database management system (`http://www.postgresql.org`). The corresponding database application for inte-

[1] Supplementary Material, can be found at `http://imsdb.mpi-inf.mpg.de`.

gration, view and analysis is based on Java SE 6. Data analysis is enabled by the integration of the free machine learning software Weka (version 3.7.3) [16]. The database model is fully included in the Java application which automatically generates the database schema. Therefore the Java code is annotated with metadata specifying the mapping between objects in Java and the relational database. This object-relational mapping (ORM), described in [17], deals with the automated persistence of Java classes to database tables. In this work, the free Java software package Hibernate (version 3.6) is used as an ORM tool (official website: `http://www.hibernate.org`). The system architecture is illustrated in Supplementary[3] Figure 1.

2.3.1 Validation and Integration

The IMSDB software tool comprises parsers, a comprehensive file and consistency validation logic, and business logic for integration of a data collection. The validation logic initially triggers the parsers to read the different files. Specification violations such as missing entries, type errors and inconsistencies for cross-related file entries are collected to present them as a comprehensive result to the end-user. The integration process is divided into three steps, where each step handles a specific kind of information: (1) IMS measurements, (2) peaks, (3) object/patient data.
Since unique constraints on entities such as the *PeakPosition* require an additional select query before inserting a new entry, we developed a method, so called chunk query peak persistence CQPP, to select multiple entities at once (depending on the chunk size) instead of executing one (or more) queries for each new entry. The insertion of missing entities is then handled in a JDBC batch process. We will see in Section 3.1 that this pays off compared to the native peak persistence method which includes a bunch of queries to insert only one peak entry with corresponding parameters.

2.3.2 Data Retrieval and Reorganization into Datasets

In a typical target dataset for analysis, peak intensities of distinct and common peak regions are arranged in columns and corresponding IMS measurements are arranged in rows. Depending on the hypothesis, only IMS measurements of particular sample objects should be included in the dataset, i.e. related labels for each instance are defined. For example, the user searches for all patients suffering from a certain disease (e.g. chronic obstructive pulmonary disease, *COPD*) to compare them to all persons which do not suffer from this disease (*control*). Thus, *control* and *COPD* are to be integrated as labels for the respective instances.
To generate such datasets by user-given attributes, two partial datasets are combined. First, a patient dataset is retrieved and restructured. The attributes of interest are arranged into a columnar format in contrast to the tabular format of the *Attribute* table. This is done by applying a dynamic pivoting (cross-tabulation) strategy utilizing the PostgreSQL function *crosstab(text, text)* of the *tablefunc* module.
The second partial dataset introduces peak information. Given the first partial dataset of objects, corresponding IMS measurements and the set of common peak regions for those measurements are extracted. A simplified query, using the hibernate query language performing this task is shown in Listing 1. The query delivers only those peak regions which are common for each

IMS measurement of the instances in the result set of patients.

Listing 1: Simplified HQL code for the retrieval of persistent *PeakRegion* entity identifiers in context with the generation of a target dataset. Given a list of IMS measurement file names, the query returns only those *PeakRegions* which are associated to all (and not less) IMS measurements in the list. The semantic of the "group by - having" statement is explained in [18].

```
SELECT r.id
FROM
  ImsMetaData i join i.peaks p join p.peakRegions r
WHERE
  i.file in :fileNameList
GROUP BY r.id
HAVING count(i) = <fileNameList.size()>
```

2.3.3 Dataset Analysis

In order to find relations between peaks and attributes such as pharmaceuticals or diseases, an automated analysis of a queried target dataset is performed. The analysis incorporates an evaluation of the classification performance of a decision tree algorithm. Standard methods (stratified ten-fold cross-validation, C4.5 decision tree algorithm [19]) from the integrated Weka library are employed after programmatic integration of any target dataset into the Weka data structures.

2.3.4 Backup, Restore and Versioning

To ensure data protection and recovery, a backup and restore system was developed. The end-user can create snapshots (backups) of the current state of the IMS database as well as restore the database to a previous state. This is ensured by triggering PostgreSQL dump and restore command line tools in the graphical user interface. Integrating and restoring data may end up in highly diverse changes. In order to track changes in the IMS database, a custom version control solution was implemented. Major changes, such as the integration of data about new IMS measurements, are logged and induce an update of the current database version.

3 Results

The IMSDB application offers a graphical user interface (GUI) which includes several windows, tabs, buttons and dialogues to provide an intuitive interaction between end-user and application. The following key features are currently supported by the GUI:

- Automated integration and validation of MCC/IMS data combined with object-attribute data.
- Presentation of database objects and their attributes, for instance patient records as shown in Supplementary[2] Figure 2.

[2] Supplementary Material, can be found at `http://imsdb.mpi-inf.mpg.de`.

- Backup and restore platform with an integrated versioning strategy.
- Dialogue-guided retrieval and automated decision tree analysis of target datasets along with decision tree visualization.

3.1 Database Upload

As explained in Section 2.3.1, we implemented the *CQPP* which reduces overhead of individual operations concerning peak data upload coupled with batch inserts. Furthermore, our system is optionally able to store the raw data of IMS measurement files into the database. The running time performance of both methods is compared on benchmarks that match typical use-cases. Five datasets of different size, increasing in the number of measurements (25, 50, 75, 100, 125) as well as the number of peak entries (3-, 6-, 9-, 12-, 15-thousand), are uploaded to a clean database. The average IMS measurements file size is approximately 8MB. The upload experiments are performed using the 64-bit database server PostgreSQL 9.0 (full logging activated) under 64-bit Windows 7 running on a Quad Core Intel Xeon CPU (2.4GHz) workstation with 24GB RAM and a SATA hard disk. Average running times of ten trials are reported in Figure 3.

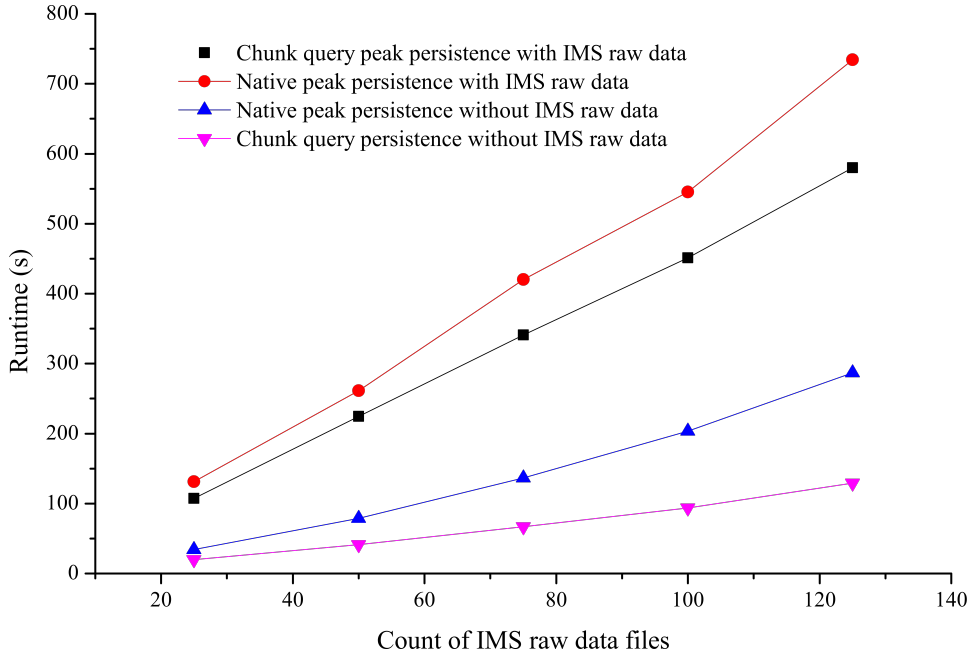

Figure 3: Report of average running times determined from ten runs of the uploading process based on five datasets with increasing number of IMS measurement files which are associated with data about 120 peak regions for each IMS measurement file respectively. Four different uploading strategies are shown. The two methods CQPP (black line, pink line) and native peak persistence (red line, blue line) are thereby combined with the options of storing the IMS raw data measurement files into the database.

3.2 Decision Tree Classification and Performance Evaluation

As a proof of concept, we used the IMS database application to analyze an anonymized real-life dataset, which includes meta information of 83 patients suffering from COPD and 35 individuals in a control group. The pre-processing of this dataset resulted in 120 peak regions, which

are evaluated for each IMS measurement. Our hypothesis in this example is that there are peaks which can be related to COPD. The attributes of interest (*COPD*, *NOT COPD*) are selected in the GUI. Subsequently, the Weka machine learning proceeds and the evaluation of the decision tree model is presented, see Figure 4. A set of classification quality measures, evaluated on the ten-fold cross validation runs, are presented: AUC (0.85), ACC (0.88), TPR (0.90), FPR (0.17) and a confusion matrix. Finally, the visualization of a decision tree which is trained on the complete dataset is triggered (see Supplementary[3] Figure 4).

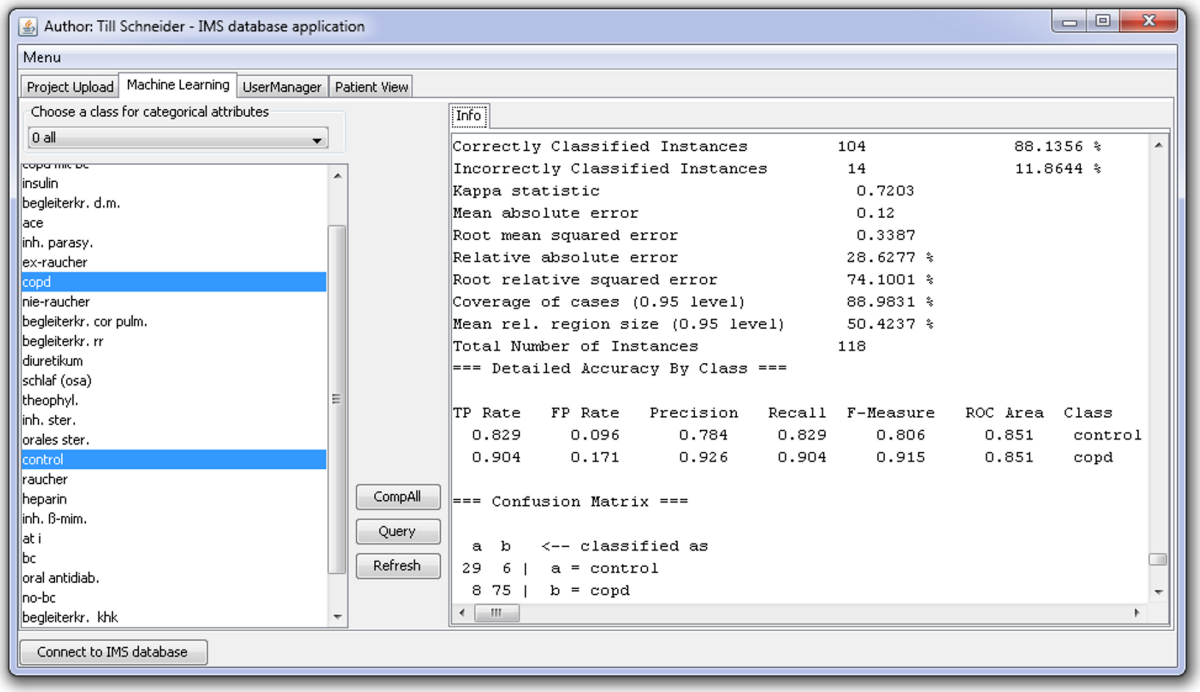

Figure 4: This snapshot of the database application presented in this work illustrates the decision tree classification performance, which is retrieved by means of a ten-fold cross-validation, for a target dataset comprising the labels (classes) *COPD* vs. *control*.

4 Discussion and Conclusion

We have presented a comprehensive database application and analysis platform that is capable of combining metabolic data, like IMS chromatograms, and heterogeneous biomedical data, e.g. patient records, into a centralized data structure. In contrast to the previous work of Lesniak [7], the ontology component enables the storage of constantly evolving object-attribute-data. In addition, the peak component ensures the fast storage and retrieval of metabolic data. As opposed to the open source database TrialDB [11], a database model was designed, not only utilizing the EAV model but also including simple classes and relations. The combination of this design and the consistency-validation- and business-logic within this application, ensures the quality of the data, the relations and the object-attribute combinations. The extension of this design towards a full EAV/CR would decrease the dependency on the application logic, but also decrease the performance of reading and writing in the system. In summary, our major contributions to the IMS biomarker community are:

[3] Supplementary Material, can be found at `http://imsdb.mpi-inf.mpg.de`.

- We provide an intuitive software system for biologists, chemists, physicists and physicians that does not require any prior knowledge or technical skills.
- Furthermore, we developed a general database model establishing the structure to combine heterogeneous biomedical and metabolic data for the purpose of research and diagnostics.
- Finally, with this paper we gave an outlook on the potential of the platform for biomarker discovery and validation by means of direct access to sophisticated statistical learning methods.

The combination of the two powerful computational tools, the extendible database as well as the machine learning toolkit, fully accessible through an intuitive graphical user interface, will accelerate and expand the opportunities of clinical diagnostic research in the near future. Our work fills a gap that hindered efficient analysis of breath-based metabolomics in biomedical research.

The supplementary material, the IMSDB database framework, the database schema and an artificial test dataset can be found at http://imsdb.mpi-inf.mpg.de.

Acknowledgements

The financial support of the Ministry of Education Science and Technology (MEST) of the Republic of Korea is acknowledged thankfully. Part of the work of this paper has been supported by the Deutsche Forschungsgemeinschaft (DFG) within the Collaborative Research Center (Sonderforschungsbereich) SFB 876, Providing Information by Resource-Constrained Analysis project TB1, Resource-Constrained Analysis of Spectrometry Data. In addition, the work was supported partly by the German Federal Ministry of Economics and Technology based on a decision of the Deutscher Bundestag within the project KF2368102AKO. We are thankful for the anonymous data sets of IMS-chromatograms related to the 3 groups as obtained by Dr. Michael Westhoff, Dr. Patrick Litterst and Barbara Obertrifter, all from Lung Hospital Hemer, Germany. JB is grateful for financial support from the Cluster of Excellence for Multimodal Computing and Interaction and the Villum Foundation. ACH is grateful for financial aid provided by the International Max Planck Research School. All authors acknowledge that parts of the presented work emerge from the MSc thesis of Till Schneider (first author) at Saarland University.

References

[1] G. A. Eiceman and Z. Karpas. *Ion Mobility Spectrometry*, volume 1. CRC Press, Taylor & Francis, 2005.

[2] M. Westhoff, P. Litterst, L. Freitag and J. I. Baumbach. Ion mobility spectrometry in the diagnosis of sarcoidosis: Results of a feasibility study. *Journal of Physiology and Pharmacology*, 58:739–751, 2007.

[3] H. Ahmadzai, D. Wakefield and P. S. Thomas. The potential of the immunological markers of sarcoidosis in exhaled breath and peripheral blood as future diagnostic and monitoring techniques. *Inflammopharmacology*, 19(2), 2011.

[4] A. C. Hauschild, T. Schneider, J. Pauling, K. Rupp, M. Jang, J. I. Baumbach and J. Baumbach. Computational methods for metabolomic data analysis of ion mobility spectrometry data – reviewing the state of the art. *Metabolites*, 2(4):733–755, 2012.

[5] J. I. Baumbach. Ion mobility spectrometry coupled with multi-capillary columns for metabolic profiling of human breath. *Journal of Breath Research*, 3(3):1–16, 2009.

[6] J. Baumbach, A. Bunkowski, S. Lange, T. Oberwahrenbrock, N. Kleinbölting, S. Rahmann and J. I. Baumbach. Ims2 - an integrated medical software system for early lung cancer detection using ion mobility spectrometry data of human breath. *J Integr Bioinform*, 4(3):75, 2007.

[7] T. Lesniak. *Entwurf, Erprobung und Bewertung eines Informationsschemas fuer Untersuchungen von Metaboliten*. Diploma thesis, University of Dortmund, Dortmund, Germany, 2007.

[8] H. Neuweger, S. P. Albaum, M. Dondrup, M. Persicke, T. Watt, K. Niehaus, J. Stoye and A. Goesmann. MeltDB: a software platform for the analysis and integration of metabolomics experiment data. *Bioinformatics*, 24(23):2726–2732, 2008.

[9] H. Neuweger, J. Baumbach, S. Albaum et al. Corynecenter - an online resource for the integrated analysis of corynebacterial genome and transcriptome data. *BMC Syst Biol*, 1:55, 2007.

[10] M. Sturm, A. Bertsch, C. Gröpl et al. OpenMS – an open-source software framework for mass spectrometry. *BMC bioinformatics*, 9(1):163, 2008.

[11] C. A. Brandt, R. Gadagkar, C. Rodriguez and P. M. Nadkarni. Managing complex change in clinical study metadata. *Journal of the American Medical Informatics Association*, 11(5):380–391, 2004.

[12] P. M. Nadkarni, L. Marenco, R. Chen, E. Skoufos, G. Shepherd and P. Miller. Organization of heterogeneous scientific data using the EAV/CR representation. *Journal of the American Medical Informatics Association*, 6(6):478 –493, 1999.

[13] W. Vautz, B. Bödeker, S. Bader and J. I. Baumbach. Recommendation of a standard format for data sets from gc/ims with sensor-controlled sampling. *International Journal for Ion Mobility Spectrometry*, 11:71–76, 2008.

[14] J. Baumbach, A. Tauch and S. Rahmann. Towards the integrated analysis, visualization and reconstruction of microbial gene regulatory networks. *Brief Bioinform*, 10(1):75–83, 2009.

[15] J. Baumbach, S. Rahmann and A. Tauch. Reliable transfer of transcriptional gene regulatory networks between taxonomically related organisms. *BMC Syst Biol*, 3:8, 2009.

[16] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I. Witten. The weka data mining software: an update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18, 2009.

[17] C. Bauer and G. King. *Java Persistence with Hibernate*. Manning Publications Co., Greenwich, CT, USA, 2006.

[18] A. Kemper and A. Eickler. *Datenbanksysteme - Eine Einführung*. Oldenbourg, München, Germany, 6th edition, 2006.

[19] J. R. Quinlan. *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993.