Quality of Life through Quality of Information
J. Mantas et al. (Eds.)
IOS Press, 2012

doi:10.3233/978-1-61499-101-4-270

Automated Realtime data Import for the i2b2 Clinical data Warehouse: Introducing the HL7 ETL cell

Raphael W. MAJEED[a] and Rainer RÖHRIG[a, 1]
[a] *Section for Medical Informatics in Anesthesiology, Justus Liebig University Gießen, Germany*

Abstract. Clinical data warehouses are used to consolidate all available clinical data from one or multiple organizations. They represent an important source for clinical research, quality management and controlling. Since its introduction, the data warehouse i2b2 gathered a large user base in the research community. Yet, little work has been done on the process of importing clinical data into data warehouses using existing standards. In this article, we present a novel approach of utilizing the clinical integration server as data source, commonly available in most hospitals. As information is transmitted through the integration server, the standardized HL7 message is immediately parsed and inserted into the data warehouse. Evaluation of import speeds suggest feasibility of the provided solution for real-time processing of HL7 messages. By using the presented approach of standardized data import, i2b2 can be used as a plug and play data warehouse, without the hurdle of customized import for every clinical information system or electronic medical record. The provided solution is available for download at http://sourceforge.net/projects/histream/.

Keywords. Clinical data warehouse, clinical decision support systems, HL7, single source secondary use, i2b2, information transformation, integration server

Introduction

Clinical data warehouses are used to consolidate all available clinical data of one or multiple organizations. They represent an important source for clinical research, quality management and controlling. Success or failure of a data warehouse depends on volume, completeness and quality of imported data, making the process of data extraction, transformation and loading (ETL) critical.

Since its introduction in 2007, the free and open source clinical data warehouse i2b2[1] quickly gathered a large user base in the research community. A single i2b2 installation consists of several server side software modules (“cells”) communicating to each other through web services. Since its introduction, i2b2 grew from its initial five core cells to a “hive” of currently 18 cells [2] plus several third party additions. Features include natural language processing, high performance computing and correlation analysis. With currently 82 related publications listed in PubMed, about 1140 publications listed in Google Scholar and more than 63000 search results in

[1] Corresponding Author: rainer.roehrig@chiru.med.uni-giessen.de

Google, it is widely accepted and commonly used for clinical data mining. Yet, little work has been done on the process of importing clinical data into i2b2 using existing standards. Typically, proprietary ETL solutions are locally developed for specific EMR implementations.

The work on this article was primarily motivated by research questions by physicians and PhD students, as well as questions regarding estimation of population sizes for clinical trials. Classically, such questions would result in the database administrator writing SQL queries for various information systems to extract exact numbers. Difficulties arise on the one hand if questions span multiple different information system with different databases. On the other hand answering of research questions takes time, especially when several questions accumulate or when the database administrator is occupied with other work. In addition to solving the basic need for a data warehouse to consolidate heterogeneous data sources, i2b2s intuitive query interface allows its frontend to be used directly by physicians and PhD students. Yet, as with any other data warehouse, most time and effort is needed for process of importing clinical data. In many cases, the introduction of a clinical data warehouse is abandoned due to difficulties with the importing ETL process (German i2b2 user group, personal communication, 2011).

Part of the know-how and software used for this project originated from a previous project to utilize HL7 messages to evaluate logic expressions and decide clinical trials' eligibility criteria [3]. Processing data parsed from HL7 messages imposed the need for short and long term storage, which can be provided by a data warehouse.

Aim of this article is to provide a reusable automated real-time data import process for the i2b2 data warehouse. A solution would serve not only to satisfy the need to merge our specific EMR implementations' data, but would also enable most hospitals using HL7 integration servers to use the free data warehouse i2b2 with little additional effort.

1. Methods

The university hospital in Gießen hosts several different information systems. Orbis (Agfa) is used for patient controlling and Swisslab (Roche) serves as laboratory information system. Most departments use Kaos (in-house development) as EMR, but different software is also in use. All information systems transfer patient information with HL7 2.x messages through an HL7 integration server. Available data include patient admission, transfers, discharge as well as all diagnoses, procedures/therapies, laboratory results and data from most medical devices.

The open source clinical data warehouse i2b2 is available for free [2]. As setup and installation of the data warehouse from scratch implies great effort and time, a ready-to-use virtual machine is provided for download. For easily replicable results, the officially provided virtual machine is used as target for the real-time import operation.

Basic and most commonly used interface for loading data into a data warehouse is a direct connection to the underlying database. The i2b2 virtual machine comes with a free but memory limited Oracle 11g express edition database server, already containing around 55000 rows of example data assigned to 133 patients. I2b2's data model is based on the "star schema"[4] which is commonly used for data warehouses. A central "fact" table contains most clinical observations, using an entity-attribute-value (EAV) approach. Each central fact is linked to a patient, an encounter, the provider and a specific concept such as lab test or diagnosis [1]. The concept in turn links into a tree-like ontology which is used by the front end to visualize and locate clinical concepts.

Development of the real-time import process is done using the Java SE development kit 6. HL7 messages are parsed using the established HL7 application programming interface for Java “HAPI” (version 1.2). Java database connectivity (JDBC) is used for database independent storage of the clinical information in the i2b2 database schema.

The real-time import process is evaluated by measuring the time required to process and import 411 MB of raw HL7 data, as it is transmitted by the integration server. In addition to the import time into i2b2’s Oracle database, a second PostgreSQL database is used to estimate import speed outside of a virtual machine. Additionally, maximum performance of the import tool is evaluated by measuring import time for a virtual database without storing any information. To determine feasibility for real-time usage, the measured times are compared to the HL7 server’s throughput statistics.

2. Results

2.1. Architecture

The proposed real-time HL7 data import module was realized using the Java SE 6 development kit. The local integration server provides all available information through a network stream using the minimal layer protocol as specified by the HL7 standard. A dedicated network port is used by the developed software to receive all messages, which are then passed through a “clean up” filter. By using regular expression search and replace operations, non-standard message segments are removed and proprietary fields are corrected. During the next step, standard conformant HL7 messages are passed to the HAPI parser which in turn provides semantically enhanced access to individual information. By using the HAPI interface, multi-valued concepts are aggregated (e.g. medication dose, route and active agent are combined into one concept) and converted to Java objects. Finally, a database layer utilizes the JDBC library to insert the semantically enhanced observations into i2b2’s Oracle database: One HL7 message contains one or more clinical facts, each of which resulting in one or more rows in i2b2’s central observation_fact table. Since encountered messages may contain clinical concepts (e.g. diagnoses, lab values) previously unknown to the data warehouse, the system recognizes new concepts and inserts appropriate rows into the concept_dimension and ontology tables. Therefore, a received fact will become available immediately to the end user through the i2b2’s query interface. The described flow of information is visualized in figure 1.

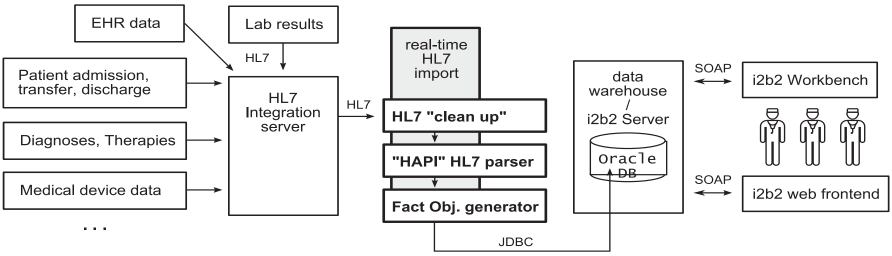

Figure 1. Flow of information from clinical information systems to data warehouse’s end users.

2.2. Benchmark

The virtual machine image provided by i2b2 already contained 55649 rows of example data in its observation fact table, assigned to 133 patients during 12386 example encounters. Due to known resource limitations of the included free Oracle database, only one week of HL7 data (411 MB) was used for benchmarking. Our dataset contained 343091 HL7 messages, consisting of 35% administrative messages (ADT), 19% diagnoses and procedures (BAR_P01) and 46% laboratory results (ORU_R01). 80% of all messages used HL7 version 2.3, followed by 18% for version 2.2, 3% for version 2.3.1 and 0.03% for version 2.5. The processing of raw HL7 messages took 8305s (~ 2 hours 18 minutes) minutes on a standard desktop computer. The measured time includes correction or removal of non-standard segments, anonymization, parsing using HAPI, aggregation of multi-valued facts and insertion into the Oracle XE database on the i2b2 virtual machine. Concepts were automatically inserted into the concept and ontology tables when encountered for the first time.

In addition to storing the information in the i2b2 virtual machine's Oracle database, all data was also transferred to a PostgreSQL database using equivalent tables. To determine the speed of the HL7 import process independent of any database, a third virtual database connection was used, which did not store anything. Results of the measurements are shown in Table 1.

Database	Time (seconds)	Speed (Kilobytes per sec.)
Oracle XE on i2b2 virtual machine (with concepts)	8270	49
PostgreSQL database	2210	186
no storage/database	388	1060

Table 1. Results of measuring the time required to process and anonymize 411 MB of raw HL7v2 messages and insert all information into the i2b2 virtual machine and a PostgreSQL database.

The total available data, including lab results and data streams from vital monitors and medical devices, amounts to around 7 Gigabytes per month. On average, 2700 bytes are transferred per second (27 KB/s).

3. Discussion

Due to known limitations of the free Oracle XE database contained in the i2b2 virtual machine, only 411 megabytes of data were used for benchmarking and feasibility analysis. Nevertheless, the average speed should be similar for larger amounts of data.

At first glance, the results presented in table 1 appear to be very poor for the Oracle database. However, the slow performance can be explained by the fact that the Oracle database is running on virtual machine which in turn is slowed down by emulation. A dedicated machine running i2b2 and the Oracle server will probably achieve speeds similar PostgreSQL. Still, first time users will most likely choose the virtual machine provided by i2b2, due to easy installation. Comparing the sub-optimal import speed of Oracle on the virtual machine of 49 KB/s to the integration server's average speed of 27 KB/s suggests that the real-time HL7 import should work for i2b2 even on slow virtual environments.

Lyman et al. developed a mapping from a clinical data warehouse to the HL7 reference information model, which might be used to export data in a standardized manner [5]. In conjunction with the present approach to import HL7 data, a complete integration of a data warehouse into HL7 based communication infrastructure is possible. The presented solution should also be generalizable to support different data warehouses other than i2b2. As long as the data warehouse implements the star schema described by Kimball [4], only small changes are required to the database-specific source code.

Real-time updates might also enable a data warehouse to be used for entirely new purposes. For example, a clinical decision support system (CDSS) could be linked to a real-time data warehouse in order to utilize large amounts of previously inaccessible data. Time critical notifications and alerts could also be triggered in conjunction with a CDSS or directly from the data warehouse.

The software developed and used for this article is available online as part of the larger project HIStream [6]. Using the provided software together with the i2b2 virtual machine, all results can be easily replicated. By using the presented approach of standardized data import, i2b2 can be used as a plug and play data warehouse, without the hurdle of customized import for every clinical information system. Existing commercial clinical data warehouses are commonly updated only once a day or even more infrequently. The presented approach allows the data warehouse to be used in real-time, as data is inserted immediately on occurrence.

References

[1] Murphy SN, Weber G, Mendis M, Gainer V, Chueh HC, Churchill S et al. Serving the enterprise and beyond with informatics for integrating biology and the bedside (i2b2). J Am Med Inform Assoc 2010; 17(2):124–30.

[2] http://www.i2b2.org/software/ (2012 Apr 16)

[3] Majeed RW, Röhrig R. Identifying patients for clinical trials using fuzzy ternary logic expressions on HL7 messages. Stud Health Technol Inform 2011; 169:170–4.

[4] Kimball R. The data warehouse toolkit: Practical techniques for building dimensional data warehouses. New York: John Wiley & Sons; 1996.

[5] Lyman JA, Scully K, Tropello S, Boyd J, Dalton J, Pelletier S et al. Mapping from a clinical data warehouse to the HL7 Reference Information Model. AMIA Annu Symp Proc 2003:920.

[6] http://sourceforge.net/projects/histream/ (2012 Apr 16)

Quality of Life through Quality of Information
J. Mantas et al. (Eds.)
IOS Press, 2012

doi:10.3233/978-1-61499-101-4-275

Coupling K-nearest Neighbors with Logistic Regression in Case-based Reasoning

Boris CAMPILLO-GIMENEZ[1,a], Sahar BAYAT[a], Marc CUGGIA[a]
[a] *Unité Inserm U936, IFR 140, Faculté de médecine, Université rennes 1, 2 Avenue du professeur Léon Bernard 35043 Rennes Cedex 9, France.*

Abstract. Case-based reasoning (CBR) systems use similarity functions to solve new problems with past situations. K-nearest neighbors algorithm (K-NN) have been used in CBR systems to define new cases status according to characteristics of past nearest cases. We proposed a new hybrid approach combining logistic regression (LR) with K-NN to optimize CBR classification. First, we analyzed the knowledge database by LR procedures and the Pearson residuals of the LR model were used to define cases' utility of the knowledge database into K-NN. Secondly, we compared the classification performances of LR model and K-NNs coupled or not with LR. Our results showed that the information provided by the residuals could be used to optimize the settings of K-NN and to improve CBR classification.

Keywords. Case-based reasoning systems; logistic models; similarity measures; k-nearest neighbors algorithms; classification.

Introduction

Case-based reasoning (CBR) is a problem-solving paradigm emerging in medical decision-making systems [1]. It utilizes previously experienced situations (cases) to solve new problems [2]. The CBR methodology is based on a four-step cycle consisting of the following processes interacting with the knowledge database (case database): retrieve, reuse, revise and retain [3]. In the medical domain, each patient can be viewed as a unique case and patient databases used in medical research can be directly exploited as knowledge databases in medical CBR systems. Similarity algorithms are used to retrieve past similar cases and have already been thoroughly described in the field of data mining [4]. The difficulty remains however in selecting appropriate settings for the similarity algorithms to ensure a "problem specific" retrieval of the most relevant cases.

In 2001, Bergmann [5] introduced the concept of « utility » in CBR, defining it as the knowledge in a database that is likely to provide the best solution to a specific problem. However the main difficulty of this approach is that utility cannot be defined before the problem is solved. Utility of past cases is therefore usually approximated by the similarity measure assessing the common characteristics between the past and the new cases. Recently, Huang [6] and Chuang [7] proposed a hybrid approach combining

[1] Corresponding author, e-mail: boriscampillo@gmail.com