University of Kentucky
UKnowledge

Theses and Dissertations--Computer Science

Computer Science

2018

INTERACTIVE CLINICAL EVENT PATTERN MINING AND VISUALIZATION USING INSURANCE CLAIMS DATA

Zhenhui Piao
University of Kentucky, kristin1105@uky.edu

Digital Object Identifier: https://doi.org/10.13023/etd.2018.260

Right click to open a feedback form in a new tab to let us know how this document benefits you.

Recommended Citation
Piao, Zhenhui, "INTERACTIVE CLINICAL EVENT PATTERN MINING AND VISUALIZATION USING INSURANCE CLAIMS DATA" (2018). *Theses and Dissertations--Computer Science*. 70.
https://uknowledge.uky.edu/cs_etds/70

This Master's Thesis is brought to you for free and open access by the Computer Science at UKnowledge. It has been accepted for inclusion in Theses and Dissertations--Computer Science by an authorized administrator of UKnowledge. For more information, please contact UKnowledge@lsv.uky.edu.

STUDENT AGREEMENT:

I represent that my thesis or dissertation and abstract are my original work. Proper attribution has been given to all outside sources. I understand that I am solely responsible for obtaining any needed copyright permissions. I have obtained needed written permission statement(s) from the owner(s) of each third-party copyrighted matter to be included in my work, allowing electronic distribution (if such use is not permitted by the fair use doctrine) which will be submitted to UKnowledge as Additional File.

I hereby grant to The University of Kentucky and its agents the irrevocable, non-exclusive, and royalty-free license to archive and make accessible my work in whole or in part in all forms of media, now or hereafter known. I agree that the document mentioned above may be made available immediately for worldwide access unless an embargo applies.

I retain all other ownership rights to the copyright of my work. I also retain the right to use in future works (such as articles or books) all or part of my work. I understand that I am free to register the copyright to my work.

REVIEW, APPROVAL AND ACCEPTANCE

The document mentioned above has been reviewed and accepted by the student’s advisor, on behalf of the advisory committee, and by the Director of Graduate Studies (DGS), on behalf of the program; we verify that this is the final, approved version of the student’s thesis including all changes required by the advisory committee. The undersigned agree to abide by the statements above.

Zhenhui Piao, Student

Dr. Zongming Fei, Major Professor

Dr. Miroslaw Truszczynski, Director of Graduate Studies

INTERACTIVE CLINICAL EVENT PATTERN MINING AND VISUALIZATION USING INSURANCE CLAIMS DATA

THESIS

A thesis submitted in partial
fulfillment of the requirements for
the degree of Master of Science in
the College of Engineering at the
University of Kentucky

By

Zhenhui Piao

Lexington, Kentucky

Co-Directors: Dr. Zongming Fei, Professor of Computer Science
and Dr. Sujin Kim, Associate Professor of Division of Biomedical Informatics

Lexington, Kentucky

2018

Copyright © Zhenhui Piao 2018

ABSTRACT OF THESIS

INTERACTIVE CLINICAL EVENT PATTERN MINING AND VISUALIZATION USING INSURANCE CLAIMS DATA

With exponential growth on a daily basis, there is potentially valuable information hidden in complex electronic medical records (EMR) systems. In this thesis, several efficient data mining algorithms were explored to discover hidden knowledge in insurance claims data. The first aim was to cluster three levels of information overload (IO) groups among chronic rheumatic disease (CRD) patients based on their clinical events extracted from insurance claims data. The second aim was to discover hidden patterns using three renowned pattern mining algorithms: Apriori, frequent pattern growth (FP-Growth), and sequential pattern discovery using equivalence classes (SPADE). The SPADE algorithm was found to be the most efficient method for the dataset used. Finally, a prototype system named myDietPHIL was developed to manage clinical events for CRD patients and visualize the relationships of frequent clinical events. The system has been tested and visualization of relationships could facilitate patient education.

KEYWORDS: Data Mining, Web Application, Clustering, Visualization, Google Chart Visualization Package

Author's signature: Zhenhui Piao

Date: June 10, 2018

INTERACTIVE CLINICAL EVENT PATTERN MINING AND VISUALIZATION
USING INSURANCE CLAIMS DATA

By

Zhenhui Piao

Co-Director of Thesis: Dr. Zongming Fei

Co-Director of Thesis: Dr. Sujin Kim

Director of Graduate Studies: Dr. Miroslaw Truszczynski

Date: June 10, 2018

ACKNOWLEDGMENTS

I represent the following thesis as my individual work. Firstly, I would like to express my sincere gratitude to my advisors Dr. Sujin Kim and Dr. Zongming Fei for their continuous support of my study and related research, their patience, motivation, and immense knowledge. Their guidance helped me during research and writing of this thesis. I could not have imagined having better advisors and mentors for my Master study.

I thank the UK Center for Clinical and Translational Science (CCTS) and High-Performance Computing (HPC) for providing the data sources and a computing environment for me to work on this research. Next, I wish to thank the complete Thesis Committee: Dr. Sujin Kim, Dr. Zongming Fei, and Dr. Tingting Yu. Each of them individual provided insights that guided and challenged my thinking, substantially improving the finished product.

Finally, I must express my very profound gratitude to my parents for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1 Introduction

With the recent development of clinical data mining technology, the challenges of diagnosing chronic diseases using large-scale data, such as insurance claims or genomic/proteomic data, have become opportunities for data-driven clinical decision-making. Chronic rheumatic diseases (CRDs), which encompass more than 200 distinct disease entities, are no exception. In particular, the high prevalence of certain rheumatic diseases, such as rheumatoid arthritis (RA), osteoarthritis (OA), vasculitis, and systemic lupus erythematosus (SLE), are associated with different pathophysiological backgrounds, including infection and autoimmune mechanisms which can benefit from clinical data mining technologies. However, attention has rarely been drawn to utilizing temporal aspects of clinical events for assisting in the early diagnosis or prediction of disease and outcomes, such as remission status within the CRD conditions. Relatedly, clinical codes have been newly implemented (e.g., ICD-10) and advanced biological findings are added constantly to diagnostic and therapeutic decision-making processes. In addition, some of the specific laboratory parameters, as well as popular use of biomedical imaging tools, like ultrasound and magnetic resonance, increasingly support the clinical decision-making process. Most importantly, effective management of the large volume of clinical data pertaining to a single patient or a disease cohort is never visualized effectively toward patient education contexts. With enthusiasm for large clinical data in research and clinical practice, mined clinical events for use in patient education are understudied in clinical data mining and data visualization. This study aims to investigate three renowned data mining algorithms for use in CRD patients education applications: FP-Growth, Apriori, and SPADE.

Once the data has been mined for sequential or association patterns, they are difficult to understand due to the technical complexing. In particular, the mined results are seldom subject to being used by patients in their understanding of disease progression. For instance, a timeline chart showing a list of clinical events in temporal order in a graphical way can be used. Some timelines work on a scale while others display diverse clinical events in sequence, including diagnosis, procedures, medication, or laboratory results. A graph can be combined with a timeline to show how quantitative data changes over time in a particular case, such as individual patient visits over time. However, there has been less attention to data visualizations using the timeline depicting clinical events. In particular, rare studies have paid attention to developing the timeline visualization for CRD patients to aid understanding of disease progression or therapeutic management in the course of treatment. This thesis aims to develop the myDietPHIL, a web visualization application for CRD patients by utilizing large insurance claims dataset. The prototype of the application has been developed and it can display CRD patients information longitudinally using a timeline view, as extracted from structured insurance claims data. Our prototype system will find a series of pattern mining results and then display the results in graphical views for patients to review. A task-based evaluation, performance and time-to-task completion was performed to measure three types of data mining techniques including FP-Growth, Apriori, and SPADE using Python and R. In addition, information overload (IO) groups are clustered into three representative levels: low, medium, and high.

The following chapter present previous related work and literature reviews. Then, our method is introduced in Chapter 3. Next, results from clustering, mining, and visualizing are included in Chapter 4. Finally, conclusions and future work are discussed in Chapter 5.

Chapter 2 Literature Review

In this chapter, prior work on clinical event mining and data visualizations as well as clinical problems in diagnosing CRDs are reviewed. This includes three sections: information overload among CRDs patients, clinical data mining, and visualization.

2.1 Information Overload and CRD Events

Information overload of health consumers has become a ubiquitous problem in modern healthcare, especially for individuals with chronic rheumatic diseases. CRDs, such as systemic lupus and vasculitis, often manifest with organ and life-threatening symptoms. Management of CRDs focuses on patient education regarding diagnosis, disease course and long-term pharmacotherapy with immunosuppression. Patients with CRDs are exposed to an endless flow of information, often at a rate far higher than their cognitive abilities can process it. Overwhelmingly, the increased adoption of personal health records (PHRs) systems as a patient version of electronic health records (EHRs), has led to an unprecedented amount of patient health information loaded in electronic format. The availability of unmanageably large and complex health information has raised concerns for information overload in CRD patients. Potential consequences of poor information management include low levels of self-care, low medication adherence, limited use of preventive services, higher rates of hospitalization, higher healthcare costs, and limited knowledge of health conditions. Ultimately, CRD patients encounter psychological pressures of information overload that potentially lead to less effective and inefficient self-care management in the long-term. To resolve the information overload issue, this study assesses which data mining algorithms better perform effective and efficient data visualization using event-mined sequences in the CRD context. Diverse personal health information management (PHIM) outcomes for CRD patients, including diagnostic, therapeutic, laboratory, and procedural codes were used to mine sequences and association patterns.

2.2 K-Means Clustering

The k-means clustering [3] is a data mining procedure which attempts to identify relatively homogeneous groups of cases based on selected characteristics, such as diagnostic events or prescription drugs. Using an algorithm that can handle large numbers of cases, this clustering algorithm requires specifying the number of clusters like the three groups of information overload in this study. The algorithm first specifies initial cluster centers (if known) or calculates any central values based on the data presented. While identifying central values of individual groups, one of two methods for classifying cases can be used for either updating cluster centers iteratively or classifying only. The results can produce cluster membership, distance information, and final cluster centers for each case and groups identified. For example, this study used k-means clustering to identify distinct groups of CRD patients whose clinical events differ in terms of information overload. That said, this study assumed that those CRD patients with more clinical event claims require more resources (e.g., time, knowledge, literacy) to process the medical information required for optimal care. The k-means cluster analysis used by this study sets up three cluster groups as distinct information overload groups. This method can identify segments of information overload for use in optimal strategy development of patient care and education in a CRD setting.

2.2.1 K-Means Clustering Algorithm

The most common algorithm uses an iterative refinement technique developed by Lloyd's algorithm [12]. Given an initial set of k-means noted in Formula 1 [12], the algorithm proceeds by alternating between two steps. The first task is the assignment step to assign each observation to the cluster whose mean has the least squared Euclidean distance, which intuitively is the nearest mean, and the second task is the update step to calculate the new means to be the centroids of the observations in the new clusters.

Formula 1:

$$S_i^{(t)} = \{x_p : \left\|x_p - m_i^{(t)}\right\|^2 \leq \left\|x_p - m_j^{(t)}\right\|^2 \forall j, 1 \leq j \leq k\}$$

$$m_i^{(t+1)} = \frac{1}{\left|S_i^{(t)}\right|} \sum_{x_j \in S_i^{(t)}} x_j$$

The formula is often presented as assigning items to the nearest cluster by distance using various distance measures such as squared Euclidean distance. This study used IBM's statistical software package called SPSS version 24.

2.3 Association Rule Mining

This thesis explored three major types of clinical event mining, including association rule minings using either FP-Growth algorithm or Apriori algorithm, and sequential mining using SPADE algorithm. Association rule mining is a well-known data mining procedure to discover how items are associated with each other among sets of items in transaction databases. A transaction database $(D = \{T_1, T_2 \ldots T_n\})$ is a set of transactions and each transaction $(T = \{I_1, I_2 \ldots I_m\})$ contains a set of one or more distinct items $(1 \leq m)$. Each transaction is then identified by unique individual identification codes with related values. Association rules have three common ways to measure patterns or data relevance, namely association. For instance, the results are set up with varying degrees of support (S), confidence (C) and lift (L). The following formula [14] shows how association rule mining is calculated.

Formula 2:

$$\text{support}(A \rightarrow B) = P(A \cup B)$$

$$\text{confidence}(A \rightarrow B) = P(B|A)$$

$$\text{lift}(A \rightarrow B) = \frac{\text{confidence}(A \rightarrow B)}{P(B)}$$

In Formula 2, Support (S) is the ratio of how frequently the items in the rule occur together. The rule $\mathrm{A} \rightarrow \mathrm{B}$ represents that the item A is antecedent and item B is consequent. The AB pairs have association in the transaction database and P means percentage of the cases containing the set AB. Confidence (C) is the ratio of both the item A and item B appearing in the same transaction. Lift (L) is the value that evaluates the quality the association rule resulted in. Sometimes, there is high support and high confidence but it may not be useful because consequent support may be higher than antecedent support. So, as above Lift (L) is needed to evaluate the quality of the mining result. For example, if there are two results including $\mathrm{support}(\mathrm{A} \rightarrow \mathrm{B})$ =0.3 and $\mathrm{confidence}(\mathrm{A} \rightarrow \mathrm{B}) = 0.75$, the rule $\mathrm{A} \rightarrow \mathrm{B}$ has high association, but if S(B) is higher than S(A), then this rule is useless. The lift ratio should be higher than 1, otherwise it is meaningless.

For instance, if there are 100 patients in transactions, 10 are diagnosed with rheumatoid arthritis (ICD-9 code: 714.0), 8 are diagnosed with osteoporosis (ICD-9 code: 733.00) and 6 are diagnosed with both of them. The $\mathrm{support}\ (714.0 \rightarrow 733.00)$ is 0.06 meaning that 6% of patients have both rheumatoid arthritis and osteoporosis. The $\mathrm{confidence}(714.0 \rightarrow 733.00)$ is equal to $\mathrm{support}(714.0 \rightarrow 733.00)/\mathrm{P}(733.00) = 0.75$. This means that if a patient record contains code 733.00 (osteoporosis), there is 75% possibility that it might also contain code 714.0 (rheumatoid arthritis). In this case, the lift is represented as $\mathrm{confidence}(714.0 \rightarrow 733.00)/\mathrm{P}(714.0) = 7.5$. Thus, the lift ratio of 7.5 is a very useful result, because the lift value over 1 means the rule $714.0 \rightarrow 733.00$ is a useful result. In other words, more patients have rheumatoid arthritis than osteoporosis in the dataset. Two association rule mining algorithms, Apriori and FP-Growth are typically used in clinical pattern mining. The following section covers more detail about these two algorithms.

2.3.1 Apriori Algorithm

The Apriori algorithm is a classic and well-known algorithm in association rule data mining. It was first introduced by Agrawal et al. in 1994 [1] for market basket data analysis. The Apriori algorithm is used to find frequent itemsets in big transaction data. The general process can be divided into two steps. The first step is to scan the database to calculate each item's support, then use a Breadth First Search(BFS) algorithm to find all the items with greater than minimum support threshold (minSup) as input. The second step uses the generated frequent itemsets to calculate candidate confidence (C), for itemsets greater than minimum confidence threshold (minCon). The following Algorithm1 shows the Apriori algorithm in pseudocode from Agrawal et al. [1].

Algorithm 1:

```
$\mathrm{Apriori}(T, \epsilon)$
$L_1 = \{\mathrm{large}\ 1 - \mathrm{itemsets}\}$
$\mathrm{for}(k = 2; L_{k-1} \neq \emptyset; k + +)$
$C_k = \mathrm{apriori} - \mathrm{gen}(L_{k-1});$
    forall transactions $t \in D$
        $c.\mathrm{count} + +;$
    end
    $L_k = \{c \in C_k \mid c.\mathrm{count} \geq \mathrm{minSup}\}$
end
return $= \bigcup_k L_k$
```

2.3.2 FP-Growth Algorithm

The FP-Growth algorithm is also a well-known association rule mining algorithm that was first introduced by Han et al., in 2004 [8]. It is designed to discover frequent patterns without using candidate generation. The algorithm compresses an input database to create a structure called a frequent-pattern tree (FP-Tree) for storing frequent items for performance improvement. Then, it divides the compressed database into branches and each branch is mined separately. The pseudocode for this approach is shown in Algorithm 2 copied from Han et al., [8].

Algorithm 2:

```
Input: FP − Tree, minimum support
Output: set of frequent patterns
FP − Growth(Tree, a){
    If tree contains a single prefix path {
        let P be the single prefix path part of Tree;
        let Q be the smultipath part with the top branching node replaced by a null ro
        for each combination $\beta$ of the nodes in the path P do
            generate pattern $\beta \cup a$ with support = minimum support of nodes in $\beta$;
            let frequent pattern set P be the set of patterns so generated;
    } else {
       let Q be Tree;
   for each item $a_i$ in Q do {
       generate pattern $\beta = a_i \cup a$ with support = $a_i$.support
       construct $\beta'$s conditional patternbase and then $\beta'$s conditional FP −
tree Tree $\beta$;
       if Tree $\beta \neq \emptyset$ then{
           call FP − Growth(Tree $\beta$, $\beta$);
        }
       let frequent pattern set Q be the set of patterns so generated;
   }
   return(frequent pattern set P $\cup$ frequent pattern set Q
                 $\cup$ (frequent pattern set P $\times$ frequent pattern set Q ))
}
```

2.3.3 Sequence Pattern Mining Algorithm

Sequence pattern mining is a special case of structured data mining and it can be classified as itemsets mining. This is typically based on association rule mining problems used to identify patterns of ordered events.

The Sequential Pattern Discovery using Equivalence classes (SPADE) algorithm is a common sequence pattern mining algorithm for discovering the set of frequent sequences. It was first introduced by Zaki et al. in 2001 [13]. The SPADE algorithm accepts vertical data. Vertical data is a set of sequences containing three fields. The first field is a sequence ID (SID), such as a customer or patient unique ID to identify the subject. The second is element ID (EID), like timestamps or visit numbers, to mark when the item occurred. The last field is for distinct items that are associated with SIDs and EIDs. In Algorithm3 below, the pseudocode shows the high level structure of the SPADE algorithm from Zaki et al., [13].

Algorithm 3:

```
$\mathrm{SPADE}(\mathrm{minSup}, \mathrm{D})$
    $F_1$ = set of frequent events;
    $F_2$ = set of frequent 2 − event − long sequences;
    for all equivalence classes $[P_i] \in F_1$ in descending order
    $E_2 = [P_i]$;
        for $(k = 3; E_{k-1} \neq \emptyset; k + +)$
            N = process_class([ε]);
            if $(N \neq \emptyset)$
                $E_k = E_k \cup N$;
            delete ⌈ε];
```

2.4 Visualization in Clinical Domain

With the introduction of large-data driven mining in clinical context, this technique has been gaining attention for the advancement of diagnostic and prognostic outcomes. In particular, for patients with complex medical problems, such as oncology patients, or chronically ill patients, or aging patients, the lack of understanding of large volumes of information can cause challenging issues like data integration and fragmentation due to longitudinal histories of clinical services received. The effective and efficient extraction of major patterns within a single patient or a group can be an effective approach to identifying critical issues in complex data driven healthcare today [4][7][10].

Chapter 3 Clinical Event Pattern Mining and Visualization

3.1 Research Questions

The goal of our work is to perform clinical event pattern mining and visualization of a University of Kentucky Healthcare dataset for CRD patients. First, the clustering of CRD patients by information overload of recorded clinical events was performed. This included five conditions: rheumatoid arthritis (RA), gout, osteoarthritis (OA), systemic lupus erythematosus (SLE), and vasculitis. We will describe the dataset used for CRD patients and their clinical events. Second, the three data mining algorithms were assessed to determine which works best in our CRD dataset in terms of efficiency and effectiveness. The efficiency was measured by processing time taken and the effectiveness was measured by best prediction of CRD clinical events. Lastly, the data mining algorithms were implemented in myDietPHIL to test two visualization techniques: timeCRD and crdMiner.

Three research questions (RQ) below were developed to assess clinical event visualization and the performance of data mining algorithms.

RQ1: Are there any variations that could be clustered into distinct groups in terms of their clinical overloads?

RQ2: Which of the data mining algorithms is most appropriate in CRD event mining within a UK HealhthCare (UKHC) dataset in terms of efficiency and effectiveness?

RQ3: Does the myDietPHIL's two visualization components perform as designed in terms of functionality and feasibility to test information overload?

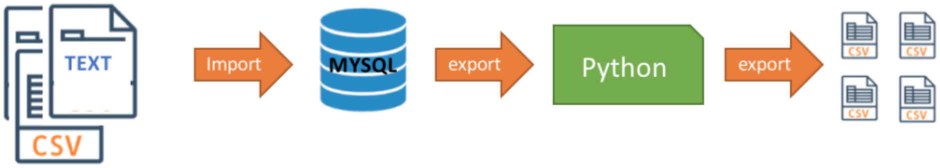
3.2 Dataset Used

We used the dataset from UK Healthcare provided by the Center for Clinical & Translational Science Enterprise Data Trust (CCTS EDT). The dataset consists of 3,289,377 rows from 12,720 distinct patients. The CCTS EDT contains clinical data from different UKHC electronic systems which has been integrated into a data warehouse and made available for the use of investigators in research projects. This dataset primarily focuses on local inpatient and outpatient data approved by the University of Kentucky Institutional Review Board (IRB). Data variables used include demographics (e.g., date of birth, gender, race), medical diagnoses (e.g., international classification of diseases ninth revision clinical modification (ICD-9)), medical procedures (e.g., current procedural terminology (CPT) codes), laboratory tests and results (e.g., chemistry, coagulation, hematology, urinalysis), and medications received (inpatient medications and outpatient prescriptions). This study did not process any other data besides clinical events and dates of claims submitted. No other social histories or demographic details were included in this analysis.

3.3 Preparation of Dataset

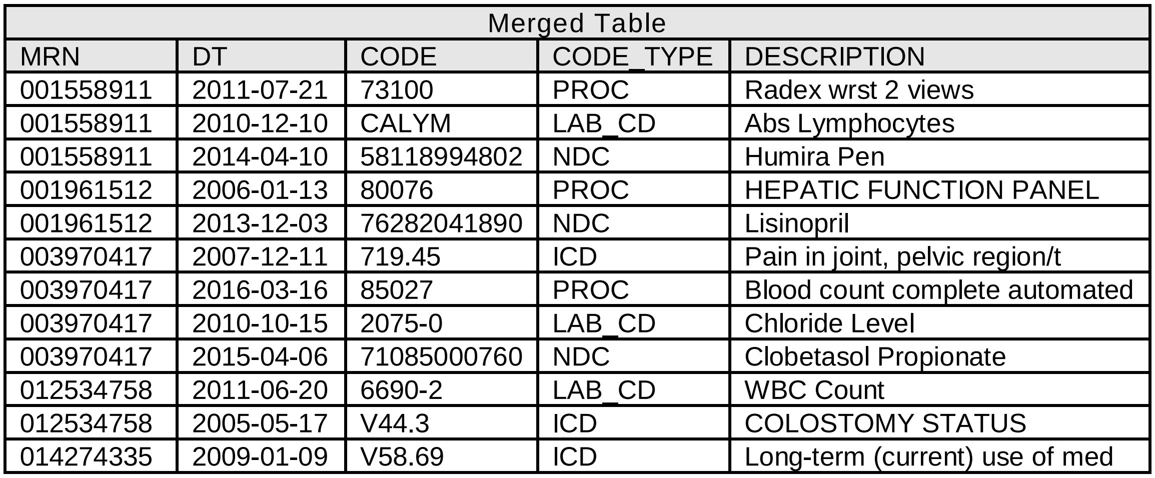
The first task was to clean up and map the dataset for use as a data mining input format. Figure 3. 1 below depicts data preparation flowchart.

Figure 3. 1: Data preparation flowchart



First, in this preprocessing step, all files that were received in .CSV formats were reformatted and then imported into the MySQL version 5.7 relational database as separate tables. Second, only eligible patient records were imported into MySQL by removing records that do not have a claim submission date or null patient identifications. In addition, some patient records that seemed to have incorrect birth years such as 2028, were removed. The third step was to unify data formats such as dates (e.g., ‘yyyy-mm-dd)’. Finally, the cleaned data were merged from four record type tables into one big table in the MySQL database. For visualization purposes, the table was re-formatted to contain essential data, such as patient unique identification, claim submission date, clinical event code, and event description. Figure 3. 2 below shows a snapshot of the big table sample merged from original small tables of four event types.

Figure 3. 2: CRD clinical event tables for use in data mining and visualization



Merged Table				
MRN	DT	CODE	CODE_TYPE	DESCRIPTION
001558911	2011-07-21	73100	PROC	Radex wrst 2 views
001558911	2010-12-10	CALYM	LAB_CD	Abs Lymphocytes
001558911	2014-04-10	58118994802	NDC	Humira Pen
001961512	2006-01-13	80076	PROC	HEPATIC FUNCTION PANEL
001961512	2013-12-03	76282041890	NDC	Lisinopril
003970417	2007-12-11	719.45	ICD	Pain in joint, pelvic region/t
003970417	2016-03-16	85027	PROC	Blood count complete automated
003970417	2010-10-15	2075-0	LAB_CD	Chloride Level
003970417	2015-04-06	71085000760	NDC	Clobetasol Propionate
012534758	2011-06-20	6690-2	LAB_CD	WBC Count
012534758	2005-05-17	V44.3	ICD	COLOSTOMY STATUS
014274335	2009-01-09	V58.69	ICD	Long-term (current) use of med

Code written in Python was used to extract each of the distinct patient event codes and associated patient IDs and also used to convert the original file format into a vertical record for use as data mining input dataset. Then, the merged big table was mapped with each items description tables for further processing.

Association rule mining using an R-package requires a basket format. The basket format must have the first column as a unique identifier of each transaction, such as patient identification or customer identification. The second column consists of items occurring in that specific transaction. For instance, we used clinical events, such as diagnostic codes (ICD-9 or ICD-10 codes), prescription medications (National Drug Code (NDC) codes), laboratory results (Logical Observation Identifiers Names and Codes (LOINC) codes), and procedural codes (Current Procedural Terminology (CPT) codes and Healthcare Common Procedure Coding System (HCPCS) codes). The codes are a combination of alphabetical and numerical values as required by character format rules of the database. And columns are separated by spaces, commas, or some other separator. This thesis intended to explore three data mining algorithms, but the input formats for three algorithms are different. For instance, Apriori requires two columns that include patient unique identification followed by unique event codes. While the input format for FP-Growth is similar to Apriori, it does not require a unique ID in the first column. Lastly, SPADE algorithm requires patient unique identification, event date, and clinical events submitted.

In addition, timeCRD and crdMiner components of the myDietPHIL application require different input data formats. Within the MySQL database, the mining result tables were then converted into visualization tables for use in myDietPHIL. This application is used to display individual patient CRD events based on mined results. Therefore, the table that is converted into the visualization view contains a list of unique events, event descriptions, and event frequency. Once the mined results were converted into a visualization table, the individual patient's events are searched and mapped onto the mined results for a visualization view.

For further classification, this study grouped the original generic clinical events into simplified classes. For diagnostic codes, this study used Clinical Classifications Software (CCS) to reduce the original ICD-9 codes down to 284 categories. The CCS is based on ICD-9, a uniform and standardized coding system. ICD-9 includes a multitude of codes - over 14,000 diagnosis codes and 3,900 procedure codes. This study further collapsed these into a smaller number of clinically meaningful categories that are sometimes more useful for interpreting data mining results. In particular, this study aimed to visualize important clinical events in a timeline view. Therefore, the simplified version was considered more appropriate. For medication classes, this study re-grouped original drug codes from NDC into NDC root classification level. The UKHC dataset included both CPT codes and ICD-9 codes for use as procedural codes. Therefore, this study converted ICD-9 into CCS procedural categories and then CPT codes were converted into HCPCS categories. For grouping laboratory data, this study used LOINC classes.

3.4 Data processing system environment

For faster and efficient data mining processing, all data conversion were processed using UK's High Performance Computing (HPC) server. Patient identifications were removed prior to HPC data processing due to non-HIPAA compliant settings in HPC. The HPC assigned to perform this study is up to 2-3 nodes with 64 GB RAM. The HPC is built on supercomputers supporting the DLX cluster. The HPC provides over 4800 processor cores, 18TB of RAM, and 1PB of high-speed disk storage supporting over 400 active UK, regional, and national researchers representing over 50 academic departments. For security purposes, this study used CCTS VM servers which are securely protected by HIPPA-compliant network requirements.

3.5 Data Mining

The following section describes the three mining algorithms that this thesis explored: FP-Growth, Apriori, and SPADE.

3.5.1 Apriori mining process

This study used R package's 'arules' library installed on HPC server. The mining jobs performed in the 'Long' partition were completed on one node. The minimum support values were re-adjusted according to mining results. For instance, myDietPHIL requires multiple types of clinical events (e.g., Diagnostic codes → Laboratory codes → Drug codes). Therefore, the Apriori results were reset to provide multiple clinical events which include frequent patterns that at least include 2-itemsets within four types of original data. Each type of minimum support value is different because this study requires at least 2 items pattern. Therefore, each event type size is different. For instance, Laboratory type has 12,326 distinct patient records from 2,584,077 event record rows. Drug code type has 1,980 distinct patient records from 30,998 event record rows. Due to a small sample size of NDC codes, the minimum support was set up for 0.005 while lab dataset was set to 0.6 minimum support value. The minimum support values were chosen to discover a large set of patterns, which then were sorted by the support statistic to identify the most frequent patterns. The minimum confidence was set to 0.8.

3.5.2 FP-Growth mining process

For FP-Growth, this study used Python 2.7 version provided by the HPC and FP-Growth algorithm implemented by Naeseth [9]. This FP-Growth in Python is different from a well-known Sequential Pattern Mining Framework (SPMF) [6] in the Java version. The Naeseth's FP-Growth was used in this study because it allows string data as an input data format. Additionally, this FP-Growth algorithm runtime is known to be faster than the Apriori. This study applied the same minimum support for both Apriori mining and FP-Growth mining so that the results could be used to compare the performance.

3.5.3 SPADE mining process

For sequential mining, this study used the R-package 'cSPADE' function within the 'arulesSequences' library installed on the UK HPC server. The cSPADE function is an R-supported C++ version of the SPADE algorithm. The input parameters require minimum support, maximum and minimum size, as well as maximum and minimum length. These mining jobs were also performed in the 'Long' partition running on one node. In order to find the optimal minimum support value, the algorithm's input parameters were run several times by adjusting different minimum supports and maximum length values. Then, the results were ordered by decreasing support to list the most frequent patterns mined.

3.6 Data Visualization

In developing the data visualization function of myDietPHIL system, the developing environment is Apache version 2.4.27, PHP version 7.1.9 and MySQL database version 5.7.19. A back-end database management is installed with version 4.7.4 PHPMyAdmin.

3.6.1 timeCRD implementation

The overarching goal of the myDietPHIL system was to develop and test to what extent health consumers might effectively improve their health conditions and information literacy by selectively managing their medical information using a new visualization tool. Within the myDietPHIL system, the timeCRD view is a horizontal timeline chart that describes patient medical history over time and further details in a summarized view. myDietPHIL draws the chart based on Google Charts timeline package. Unfortunately, Google's timeline API has some limitations, such as lack of customization of label background color, restriction of present duration, and constraints of label design that include only providing a sticky label when scrolling right to move in the timeline. The dataset used for this thesis is based on single events spanning the previous 17 years (2001-2017). Thus, we modified the Google API to better fit with the myDietPHIL design. Scalable Vector Graphics (SVG) 1.0 version [5] was used to load images on the Google timeline duration bar, so this feature can change any icons that myDietPHIL wants to represent the events.

The timeline drawing process is simple that includes PHP to post the patient ID and time range, and then fetching data from MySQL via PHP for returned query results in JSON format. In the timeline chart, the myDietPHIL uses the merged results of all four event types from a big table stored in MySQL. The Google timeline input data format is dataTable, that consists of event type, tooltip, and date. Thus, the myDietPHIL system parses returned JSON format results for importation to the dataTable created and then draws the timeline view.

3.6.2 crdMiner Implementation

The crdMiner is a topic-based knowledge map that can be used to summarize, cluster, and visualize co-occurring networks of important key events extracted from the insurance claims data. We used the mining results to identify essential information such as major diagnostic events, specific medications, procedure or laboratory findings from structured claims data extracted from UKHC electronic medical records.

The crdMiner component consists of three panels. The first panel presents the association rule mining results using an undirected graph. The second panel displays sequential pattern mining results using a directed graph. The graph API from VIS library is an open source library developed by Almende et al. [2]. It is a dynamic browser-based visualization library designed for easy use and to handle large amounts of dynamic data. Also, the VIS facilitates manipulation and interaction with big data. The last panel shows sequential pattern mining results using Google Sankey diagram API. The Sankey diagram is used for describing a flow from one set of values to another. The crdMiner's graph implementing process is similar to the process of implementing the timeline chart that fetches data via PHP and returned in JSON format. Then, the JSON format data was converted to dataTable format.

Chapter 4 Results

The three research questions described in the previous section were studied and results are reported in the following sections.

4.1 RQ 1: Three IO groups among CRD patients

The first research question was sought to address if there are any distinct groups of information overload based on the clinical event claims among five chosen chronic patients who visited UKHC Systems between 2001 and 2017. The following three tables reports the results of k-means clustering.

Table 4. 1: Demographic characteristics by three IO groups

Demographics		Three Information Overload Groups							
		Low		Medium		High		Total	
		N	% of Total	N	% of Total	N	% of Total	N	% of Total
Age	Mean (Std. Dev)	59.42	82.70%	59.74	11.50%	60.85	5.80%	59.54	100.00%
Gender	Female	5887	48.10%	803	6.60%	398	3.30%	7088	57.90%
	Male	4235	34.60%	608	5.00%	312	2.50%	5155	42.10%
Race	White	8488	69.30%	1202	9.80%	610	5.00%	10300	84.10%
	African American	1275	10.40%	143	1.20%	73	0.60%	1491	12.20%
	Others	360	2.90%	66	0.50%	27	0.20%	453	3.70%
State	KY	9630	78.70%	1335	10.90%	663	5.40%	11628	95.00%
	Others	493	4.00%	76	0.60%	47	0.40%	616	5.00%

By running k-means clustering using IBM SPSS version 24 [11], three distinct groups were formed and reported in Table 4. 1. Three groups are statistically significant, but their demographic distinction were not significant. Table 4. 1 shows a demographic characteristic of three clustered groups: low IO group (N=10,123; 82.70%), medium IO group (N=1,411; 11.50%), and high IO group (N=710; 5.80%). The low IO group is the most prevalent group and includes slightly more female than male patients. All three groups are predominated by the white racial group. This is in line with Kentucky demographics, which is relevant, as 95% of included patients were Kentuckians.

Table 4. 2 shows three IO groups by five chosen CRD conditions: RA, OA, SEL, Vasculitis, and Gout. This result indicates that a predominant proportion of each of five disease conditions belong to Low IO group, followed by Gout and Lupus patients. The OA and Vasculitis patients were distributed similarly into three IO groups. This result indicates that RA is the most prevalent disease in CRD conditions and the percentage is 30.70%.

Table 4. 2: Three IO groups by five CRD groups

IO By Disease Groups		RA	OA	Lupus	Gout	Vasculitis
Low	N Patients	3292	870	1470	1894	884
	% within Low	32.50%	8.60%	14.50%	18.70%	8.70%
Medium	N Patients	296	69	164	261	74
	% within Medium	21.00%	4.90%	11.60%	18.50%	5.20%
High	N Patients	166	26	59	122	38
	% within High	23.40%	3.70%	8.30%	17.20%	5.40%
Total	N Patients	3754	965	1693	2277	996
	% of Total	30.70%	7.90%	13.80%	18.60%	8.10%

4.2 RQ 2: Data mining results

Table 4. 3 shows most frequent patterns discovered by the Apriori algorithm from CRD patients insurance claims data. The first column represents each pattern’s support value, and the second column is the frequency count, and the last column is the top three frequent patterns from each event type. The high frequency events in diagnosis are V58.83, V58.69, both representing long-term use of medications and 714.0, M06.9, both representing Rheumatoid arthritis. The most frequent patterns in medication is 69315012710→68382077501, which represents Folic Acid and Methotrexate and are the most famous Rheumatic disease treatments. The classification level mining result shows most of patterns combineing laboratory and procedure events.

Table 4. 3: Apriori result

Support	Frequency	Most Frequent Apriori mined pattern
		Diagnosis: top 3 patterns
0.07042025	868	V58.83→V58.69
0.0545189	672	714.0→V58.83→V58.69
0.04153821	512	M06.9→V58.69→714.0
		Laboratory: top 3 patterns
0.60157391	7415	GFRB→GFR
0.60157391	7415	GFR→GFRB
0.60157391	7415	GFRB→CREA
		Procedure: top 3 patterns
0.42987235	5287	80053→85027
0.38238881	4703	80053→36415
0.3779169	4648	85004→85027
		Medication: top 3 patterns
0.12468063	244	69315012710→68382077501
0.04752172	93	68382077501→70882011730→69315012710
0.04752172	93	69315012710→70882011730→68382077501
		Classification level: top 3 patterns
0.60911894	7508	HEM/BC→CHEM
0.60911894	7508	CHEM→HEM/BC
0.55614149	6855	CTG6549→HEM/BC

Table 4. 4 shows most frequent patterns found by FP-Growth algorithm. The first column is each pattern’s frequency count value, and the second column is top three high frequency patterns from each event type. Most of high frequency patterns consist of two items. Item 2160-0 shows high frequency in the top laboratory frequent patterns. It was named CREA, representing that test of creatinine in serum or plasma belongs to CHEM class. The top three frequent patterns in procedure contain event 85027(Blood count) which was used as a screening test for various disease states and can assist in diagnosis of hematologic disorders.

Table 4. 4: FP-Growth result

Frequency	Most Frequent FP-Growth mined pattern
	Diagnosis: top 3 patterns
1294	714.0→V58.69
1008	401.9→274.9
971	714.0→401.9
	Laboratory: top 3 patterns
7415	2160-0→48642-3
7415	2160-0→48643-1
7415	2160-0→CREA
	Procedure: top 3 patterns
5093	85027→80053
4531	85027→85004
4091	85027→36415
	Medication: top 3 patterns
244	68382077501→69315012710
116	68382077501→70882011730
114	70882011730→69315012710
	Classification level: top 3 patterns
7508	CHEM→HEM/BC
6855	HEM/BC→CTG6549
6774	CHEM→CTG6549

Table 4. 5 shows result of the SPADE algorithm. The first column is each pattern’s support value, and second column is the top three most frequent sequential patterns. The single item repeat patterns in diagnosis like 714.0→714.0 on the first row indicates 14.32% patients diagnosed as RA twice and 714.0→714.0→714.0 on the third row indicates 9.76% patients diagnosed as RA three times. And pattern 68382077501→68382077501 in medication presents 6.12% patients refilled the Methotrexate medication.

Table 4. 5: SPADE result

Support	Most Frequent Sequential pattern
	Diagnosis: top 3 patterns
0.143216673	714.0→714.0
0.099410145	714.0→V58.69
0.097601258	714.0→714.0→714.0
	Laboratory: top 3 patterns
0.700265252	GFRB→2160-0
0.700265252	GFRB→48642-3
0.700265252	GFRB→48643-1
	Procedure: top 3 patterns
0.424574401	85027→36415
0.419530265	80053→85027
0.375236444	85004→85027
	Medication: top 3 patterns
0.119372787	68382077501→69315012710
0.061203844	68382077501→68382077501
0.06019221	68382009605→68382009605
	Classification level: top 3 patterns
0.604718836	HEM/BC→CHEM
0.553362170	CTG6549→HEM/BC
0.541879669	CTG6549→CHEM

Table 4. 6: Data mining runtime comparison

Dataset	minSup	Association Rule mining dataset Row	Apriori Runtime(second)	FP-Growth Runtime(second)	Sequence pattern mining dataset Row	SPADE Runtime(second)
Diagnosis generic level	0.05	12326	4.776	4.025	46829	3.426
Lab generic level	0.60	12326	9.080	0.619	68832	11.158
Procedure generic level	0.10	12299	5.122	3.149	52273	2.882
Medication generic level	0.01	1957	2.449	0.925	4443	0.123s
Classification level	0.20	12326	6.461	0.943	79774	12.693

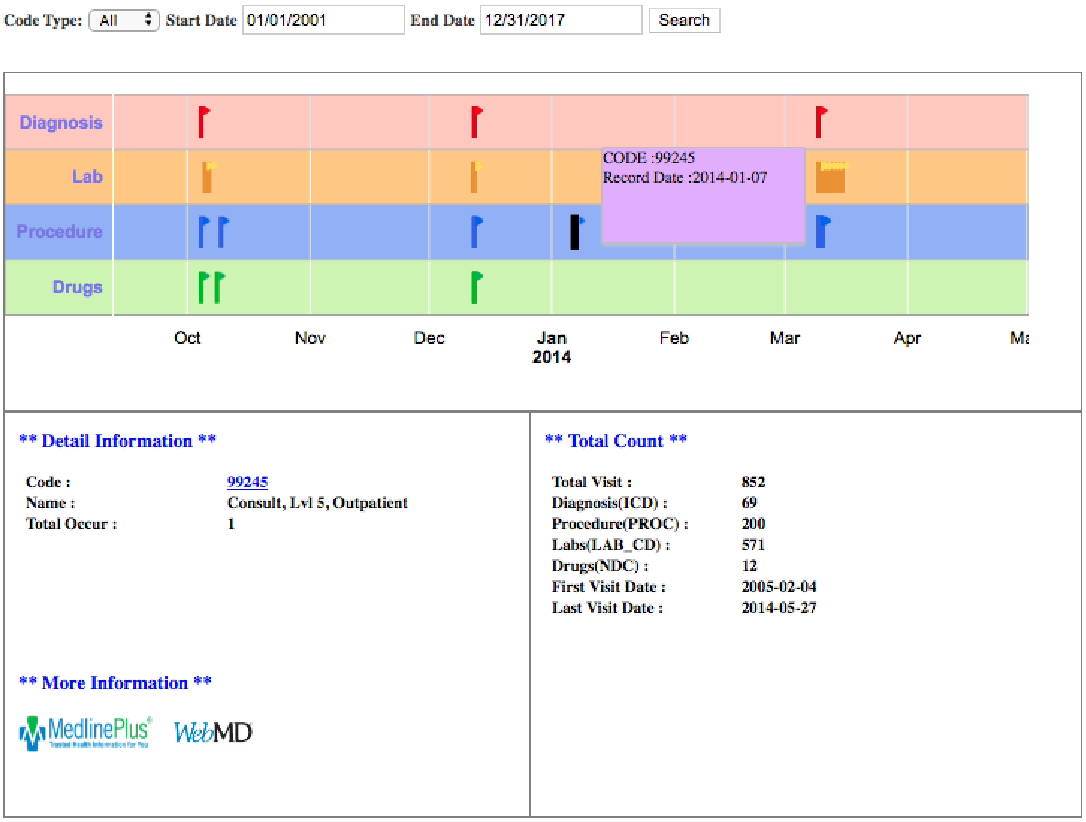
Table 4. 6 shows the performance of three mining algorithms . All algorithms were given same minimum support level for comparing the runtime. The first column is the list of minimum support value. The second column shows the number of input dataset rows used by the Apriori and FP-Growth algorithms. The third and fourth column indicates that FP-Growth algorithm is most efficient with diagnosis, lab, and classification level datasets with given same support level and inputs. The

last column indicates SPADE algorithm runs the procedure and medication datasets most efficiently.

4.3 RQ 3: A prototype system of clinical event visualization

The myDietPHIL system is designed to serve chronic rheumatic disease patients in their discovery of major clinical events based on their insurance claims data. The following screenshots show how the mined sequences are represented graphically by event patterns.

Figure 4. 1: myDietPHIL timeCRD snapshot



Code Type: All Start Date 01/01/2001 End Date 12/31/2017

CODE :99245
Record Date :2014-01-07

**** Detail Information ****

Code :	99245
Name :	Consult, Lvl 5, Outpatient
Total Occur :	1

**** More Information ****

MedlinePlus WebMD

**** Total Count ****

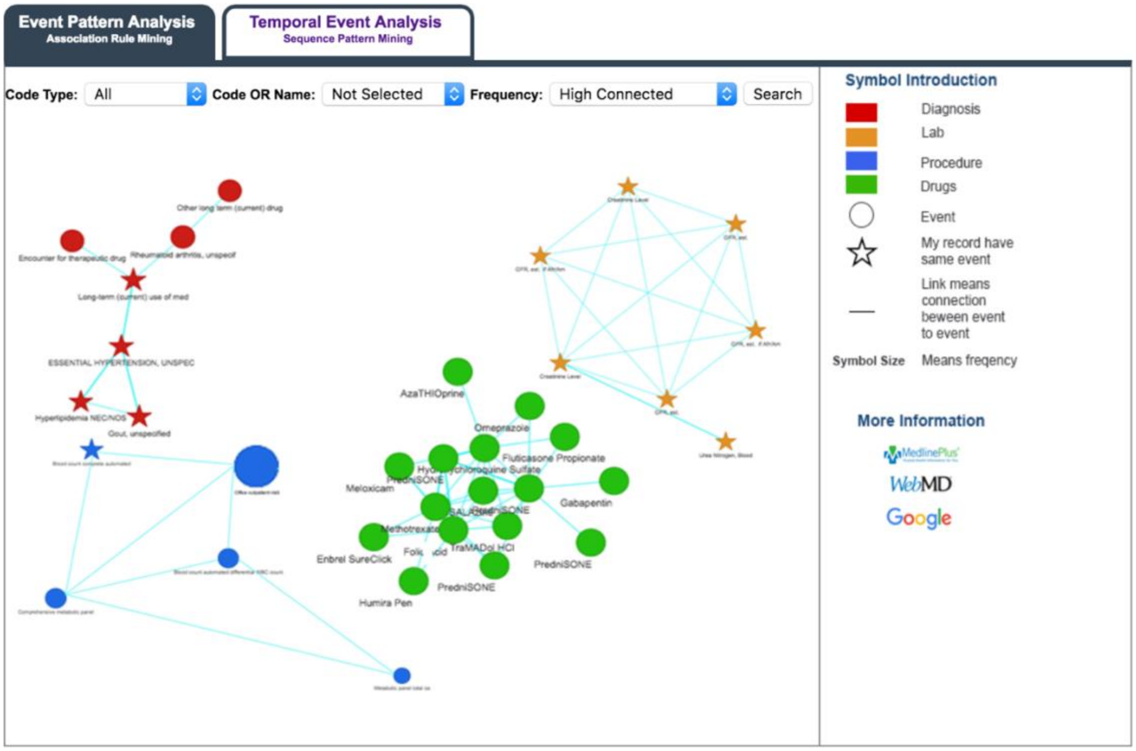
Total Visit :	852
Diagnosis(ICD) :	69
Procedure(PROC) :	200
Labs(LAB_CD) :	571
Drugs(NDC) :	12
First Visit Date :	2005-02-04
Last Visit Date :	2014-05-27

Figure 4. 1 shows the timeCRD component of myDietPHIL. The timeCRD consists of three parts, timeline chart, detail information panel, and total count panel. In timeline chart, the X-axis represents time range, while the Y-axis represents the four event types. Diagnostic events are marked as red. Laboratory events are marked as orange. Procedural events are marked as blue. Medicaition

events are coded as green. Each event is displayed as flag icon and mouse over features was implemented to show each event’s short description tooltips including clinical event codes and recorded date information. If one clicks on the flag icon, the detail information panel will display all events’ detail information occurred that day including each event’s medical code, name, occurred frequency and hyperlinks to the medical dictionary sites. The total count panel contains a snapshot summary for total event counts and the first and last date events were submitted. As a result of clinical event extraction from large insurance claims data, we were able to successfully implement timeCRD function as shown in Figure 4. 1. This system will not only be beneficial for CRD patients but also be useful for healthcare providers for their efficient and informative communication about patients’ clinical histories. In addition, this screen tracks down the clinical events within the designated timeframe, so that CRD patients’ recall and memory would be facilitated by the visual displays.

The Figure 4. 2 shows the frequent patterns discovered from each event types by FP-Growth algorithm. The circle represents each event item from data mining results. The size of the circle indicates the frequency of an event. Thus, the bigger the circle, the higher the frequency of appearance in mined clinical histories. The star represents match case. If the system finds the user's medical record contains same events with discovered frequent patterns from UKHC dataset, the system presents the event as a star icon. This result indicates the relationship between individual patient's medical histories with other patients frequent medical history patterns in a graphical way.

Figure 4. 2: myDietPHIL crdMiner association rule panel snapshot



Code Type: All
Code OR Name: Not Selected
Frequency: High Connected

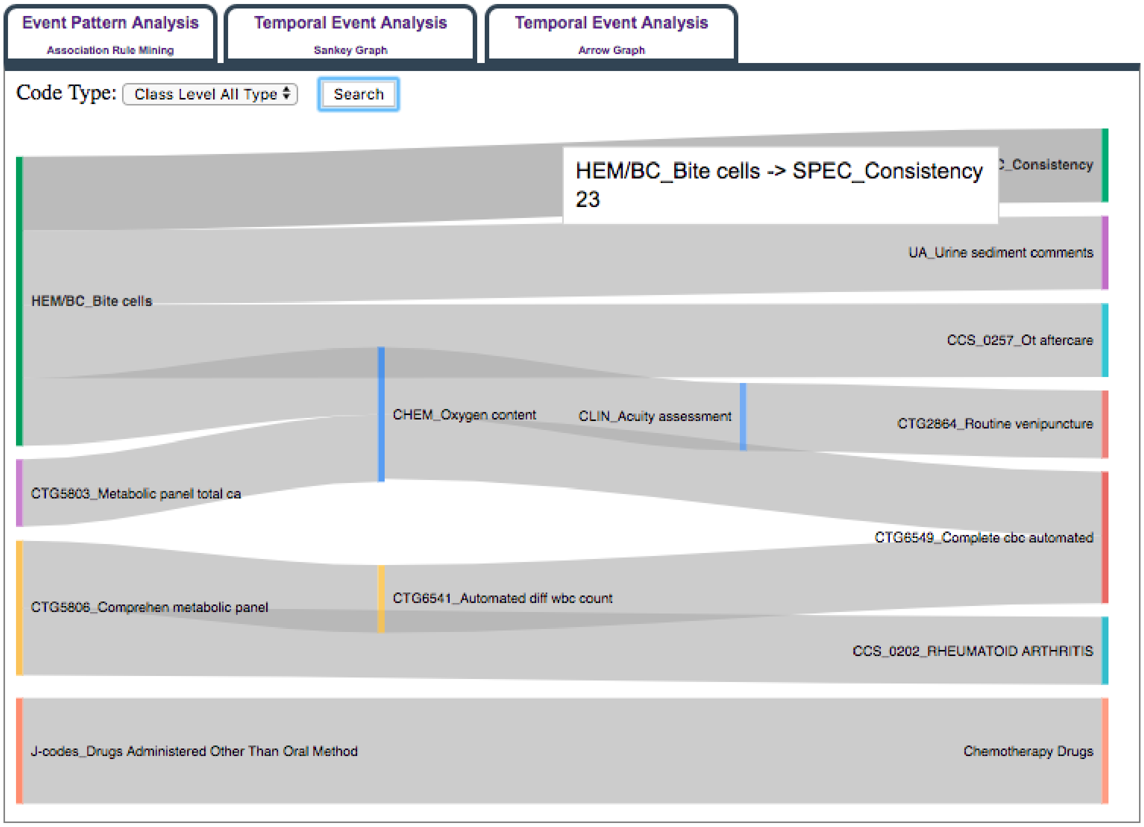
Symbol Introduction

🟥 Diagnosis
🟧 Lab
🟦 Procedure
🟩 Drugs
○ Event
☆ My record have same event
— Link means connection beween event to event
Symbol Size Means freqency

More Information

Figure 4. 3 shows classification level of frequent sequential patterns found by SPADE algorithm. The color bar represents each event found by the SPADE algorithm. The gray color links represent relationship between two events. The size of link represents the frequency of sequential patterns. The popup message indicates the frequency of connection between clinical events occurring within a designated patient population. For example, HEM/BC to SPEC patterns occurred 23 times within 12,326 patient medical records in the current dataset used. This diagram also shows predication of the next visit.

Figure 4. 3: myDietPHIL crdMiner sequential pattern panel Snapshot



Event Pattern Analysis
Association Rule Mining
Temporal Event Analysis
Sankey Graph
Temporal Event Analysis
Arrow Graph
Search
_Consistency
UA_Urine sediment comments
HEM/BC_Bite cells
CCS_0257_Ot aftercare
CHEM_Oxygen content
CLIN_Acuity assessment
CTG2864_Routine venipuncture
CTG5803_Metabolic panel total ca
CTG6549_Complete cbc automated
CTG5806_Comprehen metabolic panel
CTG6541_Automated diff wbc count
CCS_0202_RHEUMATOID ARTHRITIS
J-codes_Drugs Administered Other Than Oral Method
Chemotherapy Drugs

Code Type: Class Level All Type

HEM/BC_Bite cells -> SPEC_Consistency
23

Chapter 5 Conclusion

The purpose of this thesis was to investigate whether the data mining algorithms can be used to discover information overload and event patterns for visual displays. The target patient population was chronic rheumatic disease patients. There are three significant findings from this study. First, this study identified three major information overload groups that might be further investigated for their appropriate adoption of visual information tools, like myDietPHIL. Based on the number of clinical events these three groups encountered, this study aimed to cluster and characterize what would be the most distinct features in these groups. Distinct characteristics of the Low information overload group shows a distinct difference from Medium or High information overload groups regarding diagnostic events received and laboratory results claimed. This study assumed that higher information overload groups would receive more overload than the other groups regarding their clinical events. Interestingly, the result from this study shows two distinct results within four types of clinical events. For instance, some diagnostic events and laboratory events are highly occurring in the Low information overload group compared to the other groups. While medication claims and procedural events were less likely to occur in the Low information overload group. This is a new but striking result that needs to be further validated for clinical relevance in further studies.

Secondly, we used some association and sequential mining techniques to identify frequent patterns within the UKHC dataset. We identified that data mining results could help CRD patients in their understanding of the history of their clinical events as well as event relationships. There is also an interesting finding between mining algorithms and their application to visualization utility in the myDietPHIL application development. For example, the Apriori algorithm can process a large dataset and discovered 417,111 patterns within a classification level dataset with a small support value. However, it only gives information about the relationship between events which cannot predict next events. In other words, the Apriori algorithm could not consider time or event order in the algorithm. The sequence pattern such as R19→CHEM does not mean that Corticosteroids (R19) first occurred and then laboratory (CHEM) followed. Rather it is considered the same sequential pattern with CHEM→R19 in Apriori algorithm.

In addition, the Apriori does not consider a strict order which is covered by FP-Growth algorithm Python version in this study. The input dataset is sorted by patient ID and recorded date before transferring to FP-Growth dataset format. Therefore, 714.0→V58.69 and V58.69→714.0 are different patterns resulted in Table 4. 4. Even if the results are recognized by two patterns, the date is still unknown. If it occurred on the same day, then it is randomly ordered. Thus, FP-Growth algorithm cannot predict next events. In Table 4. 6, it does show that FP-Growth is obviously faster than Apriori algorithm in all datasets. Hence, this study used the FP-Growth algorithm results for association rule mining visualization (crdMiner) to represent relationships between frequent events.

Lastly, the SPADE algorithm considered event order, which shows prediction results in Table 4. 5. It shows high efficiency patterns with some input dataset in Table 4. 6. The event is sorted by date of occurrence. For instance, 714.0→714.0→714.0 means 9.76% of patients were recorded for 714.0 (RA) in three visits. This information is very important because if a patient got two 714.0 diagnostic codes at the previous visits, this result can also predict the possibility of third diagnostic visit. Unfortunately, the Sankey diagram cannot represent this pattern in Figure 4. 3 because of Sankey diagram limitations. A Sankey diagram is a famous diagram to perform flow relations, but it cannot scale cycle relationships nor repetitive connections.

For further study, it is advised to find a new diagram to be suitable for sequential pattern mining outputs. In addition, a unified format of simplification of clinical events by using semantic type coding would be beneficial. In addition, some new imported treatments or rare diagnoses could not be discovered by association rule algorithms in this study. Therefore, improvement of the classic algorithms to focus on discovering infrequent patterns is suggested.

References

[1] Agrawal, R., Srikant, R. (1994, September). Fast algorithms for mining association rules. 20th int. conf., Vol. 1215, pp. 487-499.

[2] Almende, B. V., Thieurmel, B. (2016). visNetwork: Network Visualization using'vis. js' Library.

[3] Alsabti, K., Ranka, S., & Singh, V. (1997). An efficient k-means clustering algorithm.

[4] Bui, A. A., Aberle, D. R., McNitt-Gray, M. F., Cardenas, A. F., Goldin, J. (1998). The evolution of an integrated timeline for oncology patient healthcare. American Medical Informatics Association AMIA Symposium, p. 165.

[5] Ferraiolo, J., Jun, F., Jackson, D. (2000). Scalable vector graphics (SVG) 1.0 specification.

[6] Fournier-Viger, P., Gomariz, A., Gueniche, T., Soltani, A., Wu, C. W., Tseng, V. S. (2014). SPMF: a Java open-source pattern mining library. The Journal of Machine Learning Research, 15(1), 3389-3393.

[7] Gotz, D., Stavropoulos, H. (2014). DecisionFlow: Visual Analytics for High-Dimensional Temporal Event Sequence Data. IEEE transactions on visualization and computer graphics, 1783 - 1792.

[8] Han, J., Pei, J., & Yin, Y. (2000, May). Mining frequent patterns without candidate generation. ACM, Vol. 29, No. 2, pp. 1-12.

[9] Naeseth, E. (2013). An implementation of the fp-growth algorithm in pure python.

[10] Perer, A., Wang, F., Hu, J. (2015). Mining and exploring care pathways from electronic medical records with visual analytics. Journal of biomedical informatics, 56, 369-378.

[11] SPSS, I. (2011). IBM SPSS statistics for Windows, version 20.0.

[12] Tang, C., & Monteleoni, C. (2016, May). On lloyd's algorithm: New theoretical insights for clustering in practice. In Artificial Intelligence and Statistics, pp. 1280-1289.

[13] Zaki, M. J. (2001). SPADE: An efficient algorithm for mining frequent sequences. Machine learning, 42(1-2), 31-60.

[14] Zhao, Y. (7 October 2016). R and Data Mining. Retrieved from http://www.rdatamining.com/

VITA

Zhenhui Piao

Education

BSc Computer Science, Yanbian University, China, June 2013

Professional Experience

Graduate Research Assistant, Biomedical Informatics Department, University of Kentucky, Lexington, Kentucky. May 2017 - Jun 2018.