Student Rating of Instruction: How Can Teachers Manage and Process the Steady Flow of Rating Data

Ngwenya, Joram
North Carolina Central University
1801 Fayetteville St, Durham, NC 27701, USA.

Abstract

At most American universities students complete an evaluation of instruction instrument at the end of each academic period. Although debate on validity and appropriateness of using students to evaluate professors and been intense and voluminous on both sides of the argument, the almost universal adoption of the instrument confirms its accepted value at most institutions. The rating instrument is administered by a central administration department and professors receive a static report of the results. Recent developments graph databases and open source R statistical programming language open opportunities for enhanced analysis of rating of results by end-users. This paper presents a methodology for transforming static reports that professors receive from a central administration department, into a graph database and from graph database to a dashboard that enables interactive analysis for scaled items and text mining for the open-ended questions.

Keywords: Graph database, data warehouse, student rating instruction, R, Neo4j, learning analysis.

1 Introduction

Debate on the validity and fairness of use of students to rate teachers and instruction they receive has raged for decades with strong arguments on both sides. Theall and Franklin [1]enumerate eight issues that have been thoroughly investigated. First, are students qualified to rate their instructors and the instruction they receive? Judging by the ubiquitous administration of a Student Rating of Instruction (SRI) at most universities, this debate is mostly settled in the affirmative. Meta-analysis of papers on the subject by different authors at different times also answers the question in the affirmative [22, 3, 10]. Second, are Ratings based Solely on popularity? A 1998 article in Chronicle of Higher Education [30] suggested this is the case although a more recent study found no relationship between average GPA and average course rating [16]. Third, are ratings related to learning? Cohen [10]found a positive relationship between ratings and learning. A recent study [11] confirms this relationship in a different way. The authors found that students with a negative perspective of a course also evaluated it negatively. Fourth, can students make accurate judgments while still in class or in school? The research appears to support that students judge instruction similarly during schooling years and after graduating [13]. The fifth question on whether students rating of instruction is reliable is answered positively in Marsh's meta-analysis of the body of work on the subject [22]. Sixth, does gender make a difference? Researchers have looked to see if there is a difference in rating female and male teachers? [12]. General finding is that there is no gender bias and when it exists appears to be explained by other situational variables. For example, female teachers being assigned junior courses which generally have less favorable ratings and rating is done by mostly freshmen. We did not find any research that examined other demographic aspects like race and ethnicity. On the popular social site, rate my professor, the heavy accent of some professors is a frequent complaint. Seventh, are ratings affected by situational variables? Although Marsh [22] concludes that ratings are robust and generally not affected by situational variables, Theall and Franklin concur but caution there are exceptions to generalizations [13]. Finally, the eight and most contentious question in SRI research is on whether students rate teachers on the basis of expected or given grades. Generally, research shows a positive relationship between higher grades and favorable course rating. To some this leads to grade inflation hence an issue to be concerned about [15] while others view this as a learning-satisfaction relationship [10].

Students that do well in a course are satisfied with the learning they received hence rate the course and professor positively. Arguments on both sides of the debate are strong, Theall and Franklin make it a question of ethics.

> Is giving higher grades in order to get higher ratings a problem with ratings or a problem with ethics, and should attempts to correct the problem be psychometric or policy issues? [16]

This problem appears to arise when ratings are considered a measurement instrument for teacher effectiveness. There are two main streams of SRI research, instrument validity, and teaching improvement. There does not appear to be any reservations on the effectiveness of ratings on teaching improvement. Studies show teachers who take rating results as formative feedback improve their ratings over time [25, 14, 27]. Teachers who take this approach would be interested in finding out the true rating of their teaching from students hence would not need to distort that input by inflating grades. We contend that teachers use SRI results for self-introspection, the question of ethics as posed by Theall and Franklin does not arise.

This paper focuses on the use of SRI for teacher self-improvement and proposes an answer to the question; How can teachers manage and process the constant flow of SRI data. We approach the question from two angles. First, a constant flow of data makes this a big data problem. When looking at results of a single period a negative trend can be easily missed because comparable past results are missing. From this first angle we consider a secondary related question; How can big data be handled at a personal level? Secondly, we consider the question from a reproducible research angle. To consistently learn from SRI results, teachers need to perform a comprehensive analysis of the results each time they are received and if trend analysis is to be performed past results need to be retrieved to be combined with the latest results. Reproducible research calls for research reports to include the data and processing methods used to produce the reported results so other researchers can replicate a study to verify the results. Similarly, comprehensive SRI analysis can be done once and reproduced automatically each time new data are added. Our proposed procedure to address the research motivation of finding a way for teachers to store SRI results data and easily perform a thorough analysis will involve three steps; First, start with SRI results report as raw input. Second, store the results in a data warehouse, and finally, retrieve data from data warehouse and analyze.

The rest of the paper is organized as follows. In the next section we review data warehousing and open-source technologies that enable data warehousing at a personal level. In section 3↓ we present and discuss a graph database model that can be used as a data warehouse for SRI results data. In section 4↓ we demonstrate how to implement our proposed solution from beginning to end using open-source technologies followed by a conclusion and suggestion for further research.

2 Data Storage Technology Review

2.1 Data Warehousing

Inman defines a data warehouse as a subject-oriented, integrated, time-variant and non-volatile collection of data in support of management's decision making process [29]. We use this definition to argue for and show usefulness of a personal database warehouse for a professor. According to the definition a data warehouse is a collection of data in support of management's decision making process. Professors meet this requirement because they are managers of the course delivery process and student evaluation of instruction is data that is used in this regard. A data warehouse for student rating of instruction is subject-oriented and the subject of interest is student evaluation of instruction. Data in a warehouse is integrated. The reports that professors received from the Office of Research and Planning can be enriched with other data at the professor's disposal. Examples include instruction materials and/or teaching methods used. Different books or learning management software could have been used in different semesters. Different sections could have been taught at different times of the day and on different days of the week. There could be a trend of rating differences between classes taught on Monday mornings and classes taught on Tuesday mornings. These differences are impossible to discern from the reports received from the Office of Research and Planning alone since those reports usually don't have the class time information. Time-variant means historical data in kept in a data warehouse. To evaluate trends in student rating of instruction, professors need historical data but often what they get are the most recent semester or current academic year survey results. A personal data warehouse can help a professor store student rating of their teaching throughout their career. This could be particularly important when a professor changes schools as their personal data warehouse can easily store ratings from different schools. Data about students rating of instruction is largely non-volatile.

Course and section information is created at the beginning of a semester and changes slowly if at all during the semester. Ratings are collected once at the end of the semester. A personal data warehouse for a professor thus meets the non-volatile definition of a data warehouse.

Kimball provides a functional definition of a data warehouse as a copy of transaction data specifically for query and analysis [26, 21]. This functional definition may in part explain why professor personal data warehouse for student rating of instruction is not a topical subject. Building a data warehouse for query and analysis is a very expensive exercise whose return on investment can be hard to quantify [4]. For this reason, data warehouses are often built exclusively at the enterprise level. The hindrance to devolving data warehouses to a personal level are thus the skills, cost and time involved, all of which most professors would probably not have.

Data warehouses are traditional implemented on relational database management systems (RDMS). A popular model is the star schema where facts to be analyzed are stored in a central Fact table. Factors to be analyzed are stored in Dimension tables linked to the central Fact table. This traditional model can be too cumbersome and complex especially for non-technical users. Different models have been proposed for specific application areas. For instance, several models have been proposed and tested for creating data warehouses for spatial data [9, 8, 6].

Users of applications build on data warehouses are mostly executives who are not technical users. Researchers are looking at ways of making data warehouses easily accessible to these non-technical users [7]. Professors whose area of specialization is not information systems or database management systems are an example of the kind of users who need easy access to warehoused data. In this paper we treat teachers as non-technical users and propose a graph database warehouse that is intuitive to them.

The original purpose of a data warehouse as defined by Inman was for storing historical data. New developments in database technology have led to a call for organizations to consider real-time data warehousing as a powerful technique for achieving operational business intelligence (BI) [5]. The graph database we propose is near real-time in the sense that when new SRI results are entered analysis is updated immediately.

2.2 Graph Database

In a relational database, information is stored in related tables. Each table stores information pertaining to a single entity and within a table rows represent an instance of an entity represented by the table. The attributes or properties of a record are represented as columns of the table, and one or more of the attributes that uniquely identifies the record is flagged as the primary key. To establish relationship between tables a primary key from one table is entered in another table as a foreign key. To extract information from a relational database, a structured query language (SQL) is used to join tables and extract required information. Queries can be complex hence trained SQL specialists are required to extract information for users either on a per request basis or through a front-end application that shields the user from complex queries. In a graph database the rows found in a relational database are represented as nodes and instead of keys linking two records an edge between two nodes represents the relationship between the nodes. Both nodes and edges are named and can have properties to further describe them. Graph databases are by design suitable for personal data warehouses unlike the traditional relational databases that are schema dependent and hence require experts to setup and operate.

NoSQL databases, especially graph databases are gaining popularity among developers as they promise delivery of superior performance [29]. Researchers are reporting successful conversions from traditional relational databases to graph databases or testing new applications that were either too difficult to implement or were not possible on Relational Database Management System (RDBMS) platforms. Examples include the significant improvements in the handling of high-resolution remote sensing imagery [3], answering why-not questions that are near impossible with relational databases [31], retrieving computational models in biological systems [32, 1, 2].

3 Graph Database Model

Figure 1↓ shows the graph database model for storing SRI survey results data. Node labels are shown outside the node and the properties of the node inside the node circle. The model allows for storage of basic SRI data that can be entered at different times. The Course nodes and nodes for the two types of SRI instrument items; open-ended and scaled questions, are semi-permanent information that can be entered once and will rarely change. Section and semester information can be added at the beginning of each semester. Response nodes for both open-ended and scaled items are entered when SRI results are received.

For open-ended items, the ShortAnswer node shows all the different answers/suggestions that students entered and the for SRIQuestion there is a link to each of the responses received. For instance, if for a particular item nine students selected the Strongly Agree option and two students selected the Disagree option, there would be nine nodes of the same color say green for the StronglyAgree nodes and two nodes of another color say red for the Disagree nodes. The section node can point to other nodes that would provide more information to enrich the analysis. Information like enrollments can be linked to the section node thus giving the professor ability to investigate questions like; does cohort composition affect overall rating or rating of a specific question?

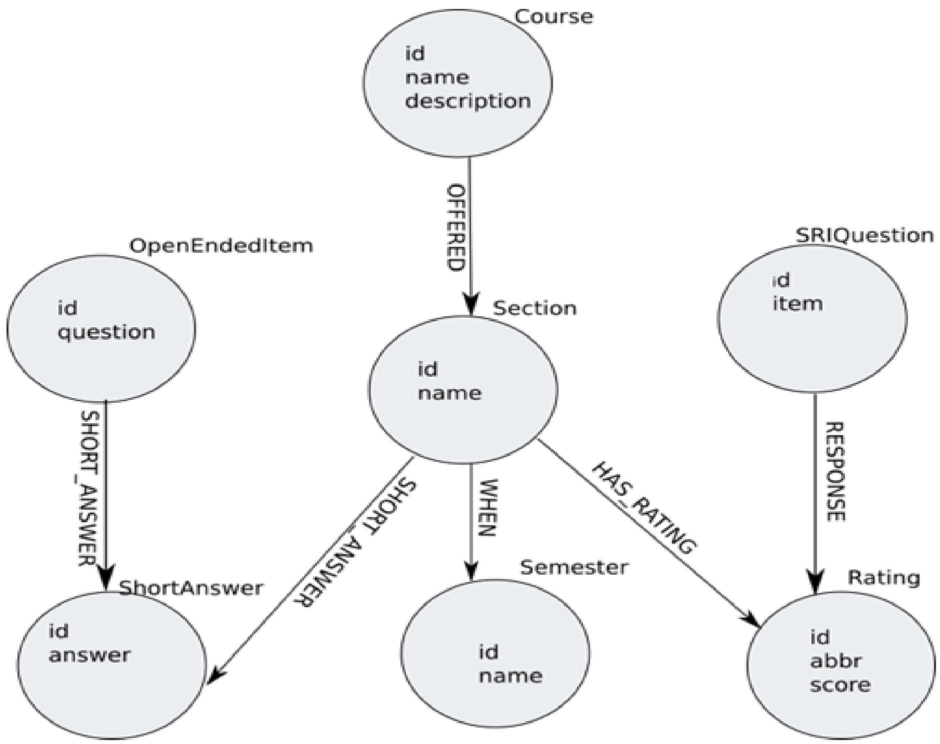

Figure 1 SRI results graph database model

4 Case Illustration

4.1 Data Extraction, Transformation, and Loading (ETL)

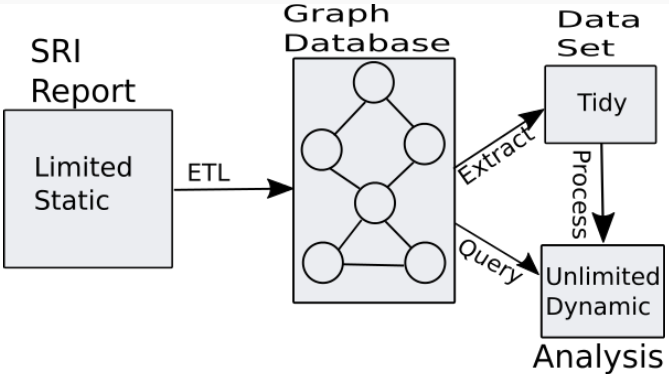

Figure 2 SRI Data Management and Analysis steps

2↑shows basic steps we propose for managing and analyzing SRI data. The demonstration is built on open-source software so anyone interested can replicate the methodology. We used the spreadsheet program LibreOffice Calk to prepare data for loading into a graph database. Data presentation for human use is sometimes not tidy for statistical analysis [19].

One common problem is when, to make the data readable to human beings, values are entered as column headings. To process the SRI survey data, the untidy data set derived from the official SRI survey reports was transformed into a tidy data set using R packages dplyr [17], plyr [18] and tidr [20]. Figure 3↓ shows the scaled items for one of four sections that were analyzed. The prepared data sets were then loaded into the graph database Neo4j [23] from RStudio using the Rneo4j [24] package following a three-step procedure. First, create a cypher query that accepts parameters. Second, create a batch of Neo4j statements by looping through a data set and supplying values to the query statement in first step with values from the data set. See R code below. The batch of statements created are treated as a single Neo4j transaction. All statements in a transaction must execute properly for the transaction to be committed. Third, commit the transaction. Five batches (Neo4j transactions) were created to load data for each of the possible responses; strongly agree (SA), agree (A), neutral(N), disagree(D) and strongly disagree(SD). For each input row, if the number of responses was not zero, nodes of the type of response equal to the number of responses were created, each linked to the question and section indicated in the row. For instance if the row indicated that nine students in section 01 answered strongly agree to question 4, then nine strongly agree nodes were created each linked to question 4 and section 01. Each node included a score property that stores the weight of the node, from 5 for StronglyAgree to 1 for StronglyDisagree. This has the effect of unwinding or untangling originally summarized data to finer granular level enabling detailed analysis. This process creates live data from originally dead data. Two batches were similarly created for loading data for open-ended questions and question nodes.

R code that was used to load SRI items into graph database.

```
# Step 1, create query statement
SA.query <- “
MATCH(s:Section{crn:{cn}})
MATCH(q:SRIQuestion{qnid:{qn}})
CREATE (q)-[:RESPONSE]->(:StronglyAgree{score:5,abbr:’SA’})<-[:RESPONSE]-(s) “
SA.t <- newTransaction(graph)
# Step 2 create transactions batch
for(i in 1:nrow(responses)){
    if(responses[i,]$SA > 0){
        for(j in 1:responses[i,]$SA){
            cn = responses[i,]$crn
            qn = responses[i,]$qnid
            appendCypher(SA.t, SA.q, cn=cn, qn=qn)
} }}

#Step 3 commit transaction
commit(SA.t)
```

Example of SRI report that was processed.

Business School SRI Survey		[redacted]							
		Responses (%)					Individual		
		SA	A	N	D	SD	N	Mean	Std Dev
Q4	The stated goals and objectives for the course are consistent with what was actually taught.	6 50%	5 41.7%	0	1 8.3%	0	12	4.3	.85
Q5	The instructor clearly presents his/her subject matter.	5 41.7%	4 33.3%	1 8.3%	2 16.7%	0	12	4.0	1.08
Q6	The subject matter of this course is well organized.	7 58.3%	4 33.3%	0	1 8.3%	0	12	4.4	.86
Q7	The instructor is enthusiastic and arouses interest in this course.	4 33.3%	4 33.3%	1 8.3%	3 25%	0	12	3.8	1.16
Q8	My power to think, criticize, and or create have been improved as a result of this course.	5 41.7%	3 25%	3 25%	1 8.3%	0	12	4.0	1
Q9	The tests and other readings assigned for this course have been helpful.	5 41.7%	5 41.7%	1 8.3%	1 8.3%	0	12	4.2	.90
Q10	The instructor uses instructional approaches (for example, discussions, lectures, audio-visuals, field work, demonstrations, computer programs, etc.) which effectively enhance learning in this course.	7 58.3%	5 41.7%	0	0	0	12	4.6	.49
Q11	The examinations are consistent with the course objectives and the instruction.	7 58.3%	4 33.3%	0	1 8.3%	0	12	4.4	.86
Q12	Quizzes, examinations and/or written assignments are provided frequently enough to help me evaluate my progress.	6 50%	5 41.7%	0	1 8.3%	0	12	4.3	.85
Q13	The instructor is genuinely concerned with students' progress.	7 58.3%	4 33.3%	1 8.3%	0	0	12	4.5	.65
Q14	I am able to get help from the instructor when I need it.	8 66.7%	3 25%	1 8.3%	0	0	12	4.6	.64
Q15	This instructor is effective in promoting learning.	6 50%	5 41.7%	1 8.3%	0	0	12	4.4	.64

Figure 3 Example of SRI Single Section report

4.2 Graph Database Analysis

The graph database can be queried and insights gained from visualization of the graph results. Figure 4↓shows results of a queries for all the scaled item responses for four section, two from each of two courses that a professor taught in a semester. The original report was a long scrolling page impossible to fit on a single screen. At a glance the professor is able to make a visual comparison of how the four sections responded. In this case, the section in to right had the most number of disatisfied responses.

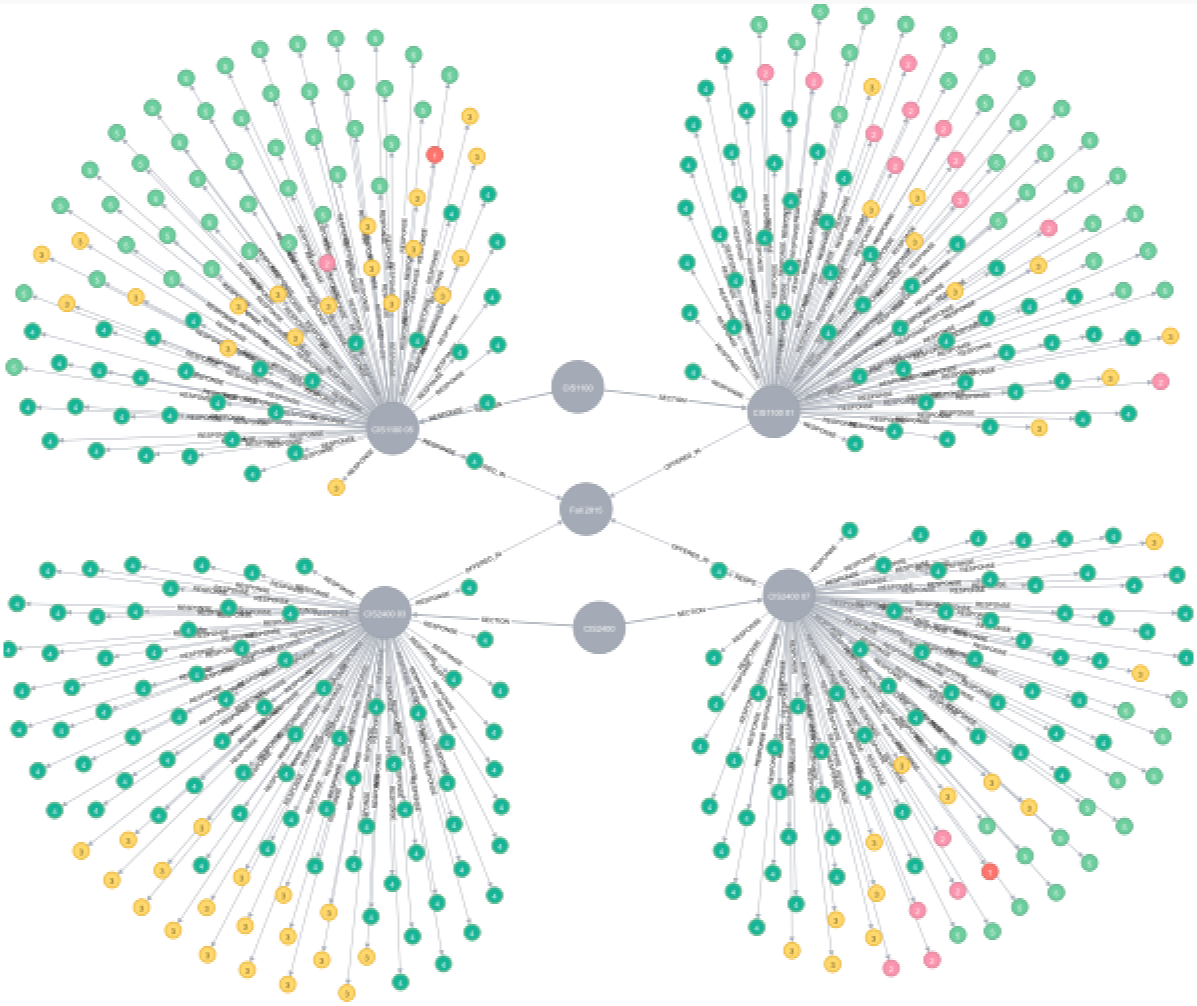

Figure 4 Example of graph database showing responses to scale items.

4.3 Statistical Analysis - Dashboard

Different kinds of data sets can be extracted from graph database and analyzed anyway the professor wishes. Extracted data sets would be already statistically tidy. Extracting a data set involves traversing all the desired nodes from the database graph and returning properties from the nodes and/or edges. If two or more edges start from the same node in the same direction an optional match clause is used to attach branching paths to the main path. This is equivalent to outer join clauses in SQL. The data set for the rating items was created by traversing the main path Course->Section->Rating<-SRIQuestion and a branch from Section->Semester. Direction of the edge can be omitted in the query. Generating an equivalent data set from a relational database would have involved multiple join statements that can be too complex for people who are not SQL specialists. The cypher query for extracting rating question data set is shown below. A similar query was used to extract data set for open-ended questions.

```
query <- "
MATCH(c:Course)--(s:Section)-[r1:RESPONSE]->(n)<-[r2:RESPONSE]-(q:SRIQuestion)
OPTIONAL MATCH (s)--(term:Semester)
RETURN term.name as semester,c.courseid as course,s.name as section,s.crn as crn,
n.abbr as abbr,n.score as score ,q.qnid as qnid, q.sortid as question
ORDER BY semester,course, section, question, score
"
df <- cypher(graph,query)
```

We created a dashboard application with R package ShinyDashboard [31] to demonstrate analysis of the SRI results.

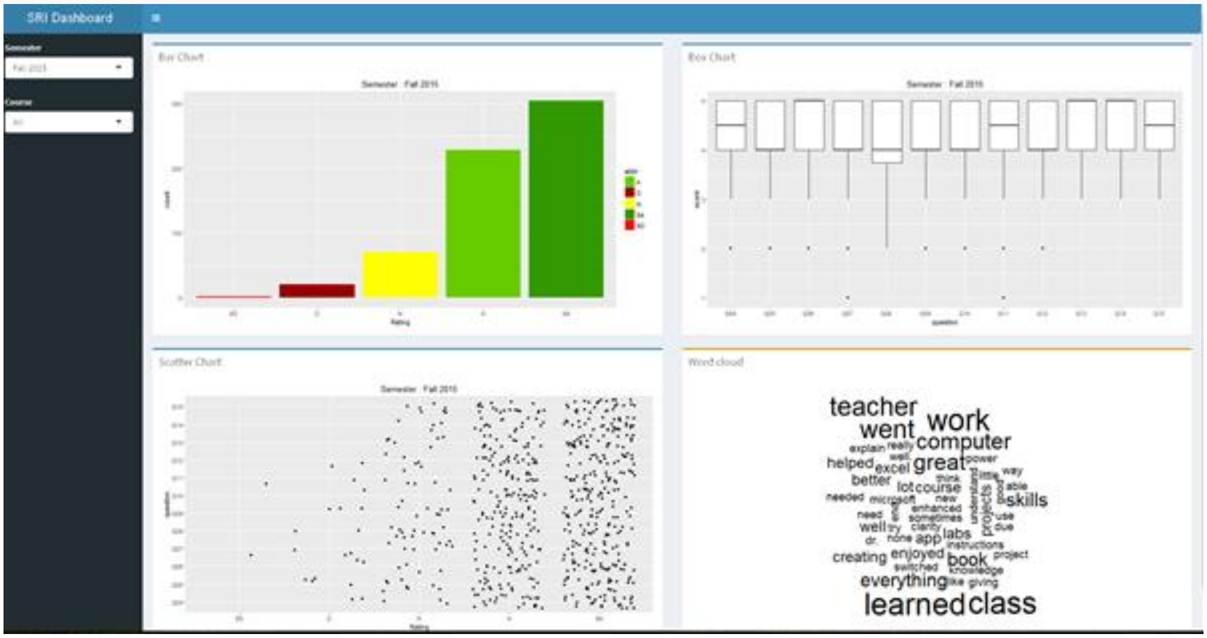

Figure 5 Dashboard Screen Shot

Figure 5↑shows a screen shot of the analysis dashboard. It demonstrates slicing and dicing analysis. Whereas the graph database was populated with data from static reports at the course section module, the dashboard enables drill down analysis from semester through the course to section. At each level, the user is presented three analytic plots and one text mining plot. The first data analytic plot is a bar chart showing a count of responses for each response type. The second is a box plot for the spread of scores for each item and the third is a rating by item scatter plot. The scatter plot is created by applying a jitter function that randomly disperses values that would otherwise pile on one spot. The word cloud plot is a text mining technique that intelligently displays words according to the frequency in the analyzed text corpus. More analysis can be added without modifying data sets. For instance, a comparison tab can be added to each of the four plot areas providing comparisons that are difficult to make from discreet static reports. Selecting all from the Semester drop down box on the left, the main plots would show data for all semesters in the data warehouse and the comparison tab would show trends over time. Selecting a semester would show comparisons by course if a professor taught multiple courses that semester. At the course level, the comparison would be by section.

5 Conclusion and Future Research

Many research papers on SRI have included other information like student grades, student characteristics, instructor gender, time of ratings. When applied at personal level comparisons across time, courses, sections and questions are critical. Static reports which are at section level, the generic format, only offer comparison by question. This paper has presented a methodology that harness new advances in data storage and analysis to easily add comparisons and add levels that are missing in static section level reports. Single page dashboard presentation makes more information available and easier to understand, than is possible with long scrolling page where some information can be hidden from view making comparisons impossible. Graph databases are schema-less. This makes it easy to add more data to the SRI results hence enriching SRI research. Our plan for future research is using this potential to graph warehouse and enrich SRI data to join the long established quest to approve or disapprove existing biasing factors in SRI research [28].

References

[1] Davide Alocci, Julien Mariethoz, Oliver Horlacher, Jerven T. Bolleman, Matthew P. Campbell, Frederique Lisacek: "Property Graph vs RDF Triple Store: A Comparison on Glycan Substructure Search.", *PLoS ONE*, pp. 1 - 17, 2015. URL http://www.nclive.org/cgi-bin/nclsm?url=.

[2] Lee Chun-Hee, Chung Chin-Wan: "Efficient search in graph databases using cross-filtering.", *Information Sciences*, pp. 1 - 18, 2014. URL http://www.nclive.org/cgi-bin/nclsm?url=.

[3] Thomas J. Lampoltshammer, Stefanie Wiegand: "Improving the Computational Performance of Ontology-Based Classification Using Graph Databases.", *Remote Sensing*, pp. 9473 - 9491, 2015. URL http://www.nclive.org/cgi-bin/nclsm?url=.

[4] Nabila Berkani, Ladjel Bellatreche, Selma Khouri: "Towards a conceptualization of ETL and physical storage of semantic data warehouses as a service.", *Cluster Computing*, pp. 915 - 931, 2013. URL http://www.nclive.org/cgi-bin/nclsm?url=.

[5] Farrah Farooq: "The data warehouse virtualization framework for operational business intelligence.", *Expert Systems*, pp. 451 - 472, 2013. URL http://www.nclive.org/cgi-bin/nclsm?url=.

[6] Luca Leonardi, Salvatore Orlando, Alessandra Raffaetà, Alessandro Roncato, Claudio Silvestri, Gennady Andrienko, Natalia Andrienko: "A general framework for trajectory data warehousing and visual OLAP.", *GeoInformatica*, pp. 273 - 312, 2014. URL http://www.nclive.org/cgi-bin/nclsm?url=.

[7] F. Majeed, M. Shoaib: "A NATURAL LANGUAGE BASED RETRIEVAL SYSTEM FOR THE DATA WAREHOUSE.", *Pakistan Journal of Science*, pp. 426 - 434, 2013. URL http://www.nclive.org/cgi-bin/nclsm?url=.

[8] Igor Timko, Curtis Dyreson, Torben Pedersen: "A probabilistic data model and algebra for location-based data warehouses and their implementation.", *GeoInformatica*, pp. 357 - 403, 2014. URL http://www.nclive.org/cgi-bin/nclsm?url=.

[9] Ganesh Viswanathan, Markus Schneider: "Querying Cardinal Directions between Complex Objects in Data Warehouses.", *Fundamenta Informaticae*, pp. 177 - 202, 2014. URL http://www.nclive.org/cgi-bin/nclsm?url=.

[10] P.A. Cohen: "Student rating of Instruction and Student Achievement: A Meta-Analysis of Multisection Validity Studies", *Review of Educational Research*, pp. 281-309, 1981.

[11] Tyrone. Donnon, Tanya Beran: "Student and teaching characteristics related to ratings of instruction in medical sciences graduate programs.", *Medical Teacher*, pp. 327 - 332, 2010. URL http://www.nclive.org/cgi-bin/nclsm?url=.

[12] K.A. Feldman: "College Students Views of Male and Female College Teachers, Part1: Evidence from Social Laboratory Experiments.", *Research in Higher Education*, pp. 317-375, 1992a.

[13] P.W. Frey: "Validity of Student Instructional Ratings. Does Time Matter?", *Journal of Higher Education*, pp. 327-336, 1976.

[14] Clinton Golding, Lee Adam: "Evaluate to improve: useful approaches to student evaluation.", *Assessment & Evaluation in Higher Education*, pp. 1 - 14, 2016. URL http://www.nclive.org/cgi-bin/nclsm?url=.

[15] A.G. Greenwald, G.M. Gillmore: "Grading Leniency Is a Removable Contaminant of Student Ratings.", *American Psychologist*, pp. 1209-1217, 1997.

[16] Tyler J. Griffin, John Hilton III, Kenneth Plummer, Devynne Barret: "Correlation between grade point averages and student evaluation of teaching scores: taking a closer look.", *Assessment & Evaluation in Higher Education*, pp. 339 - 348, 2014. URL http://www.nclive.org/cgi-bin/nclsm?url=.

[17] Hadley Wickham, Romain Francois: *dplyr: A Grammar of Data Manipulation*. 2015. URL https://CRAN.R-project.org/package=dplyr. R package version 0.4.3.

[18] Hadley Wickham: "The Split-Apply-Combine Strategy for Data Analysis.", *Journal of Statistical Software*, pp. 1-29, 2011.

[19] Hadley Wickham: "Tidy Data", *Journal of Statistical Software*, 2014.

[20] Hadley Wickham: *tidyr: Easily Tidy Data with `spread()` and `gather()` Functions*. 2016. URL https://CRAN.R-project.org/package=tidyr. R package version 0.4.1.

[21] Warren Thorntwaite Joy Mundy: *The Microsoft Data Warehouse Toolkit*. Wiley, 2006.

[22] H.W. Marsh: “SStudents' Evaluations of University Teaching: Research Findings, Methodological Issues, and Directions for Future Research”, *International Journal of Educational Research*, pp. 253-388, 1987.

[23] Neo4: *Neo4j Database*. Electronic, URL http://www.neo4j.org.

[24] Nicole White: *RNeo4j: Neo4j Driver for R*. 2016. URL https://CRAN.R-project.org/package=RNeo4j. R package version 1.6.3.

[25] Stuart Palmer: “The performance of a student evaluation of teaching system.”, *Assessment & Evaluation in Higher Education*, pp. 975 - 985, 2012. URL http://www.nclive.org/cgi-bin/nclsm?url=.

[26] Ralph Kimball, Margy Ross.: *The Data Warehouse ToolkToolkit: The Definitive Guide to Dimensional Modeling*. John WWiley & Sons, 2013.

[27] Mary Ryan: “Framing student evaluations of university learning and teaching: discursive strategies and textual outcomes.”, *Assessment & Evaluation in Higher Education*, pp. 1142 - 1158, 2015. URL http://www.nclive.org/cgi-bin/nclsm?url=.

[28] M. Theall, J. Franklin: “Looking for Bias in All the Wrong Places: A Search for Truth or Witch Hunt in Student Ratings of Instruction”, *New Directions for Institutional Research*, pp. 45-56, Spring 2001.

[29] W.H. Inmon: *Building the Data Warehouse*. Wiley, 2002.

[30] Wilson R.: “New Research Casts Doubt on the Value of Student Evaluations of Professors”, *Chronicle of Higher Education*, pp. A1,A16, 1998.

[31] Winston Chang, Joe Cheng, JJ Allaire, Yihui Xie, Jonathan McPherson: *shiny: Web Application Framework for R*. 2016. URL https://CRAN.R-project.org/package=shiny. R package version 0.13.2.