

An Alternative Approach to Teaching Bigdata and Cloud Computing Topics at CS Undergraduate Level

Debzani Deb

Department of Computer Science
Winston-Salem State University
Winston-Salem, NC, USA
debd@wssu.edu

Muztaba Fuad

Department of Computer Science
Winston-Salem State University
Winston-Salem, NC, USA
fuadmo@wssu.edu

Keith Irwin

Department of Computer Science
Winston-Salem State University
Winston-Salem, NC, USA
irwinke@wssu.edu

ABSTRACT

Big data and cloud computing collectively offer a paradigm shift in the way businesses are now acquiring, using and managing information technology. This creates the need for every CS student to be equipped with foundation knowledge in this collective paradigm and to possess some hands-on-experience in deploying and managing big data applications in the cloud. We argue that, for substantial coverage of big data and cloud computing concepts and skills, the relevant topics need to be integrated into multiple core courses of undergraduate CS curriculum rather than creating additional standalone core or elective courses. Our approach to including these topics is to develop learning modules for specific core courses in which their coverage might find an appropriate context. In this poster, three such modules are discussed and our classroom experiences during these interventions are documented. Our objective is to share our experience with the academics who aim at incorporating similar pedagogy and to receive feedback about our approach.

KEYWORDS

Bigdata, Cloud Computing, Curriculum

1. Introduction

We believe that every undergraduate CS student should be equipped with foundation knowledge in the collective paradigm comprised of big data analytics and cloud computing and should possess some hands-on experience in deploying and managing big data applications in the cloud to acquire skills that are necessary to meet current and future industry demands as well as to enable them to carry out applied research on this paradigm. However, the challenge is that many of the tools and techniques of the big data and cloud computing paradigm have emerged only in the last few

years and have not yet transitioned into the most recent ACM/IEEE Joint Curriculum recommendations [1] or the ABET curriculum requirements [2]. Many 2-year and 4-year institutions develop their CS curriculum around these guidelines and requirements and as a result cannot afford to include these contemporary topics as required (core) courses in their densely packed curricula. A number of institutions are now offering non-core specialized courses [3, 4, 5] to cover a variety of aspects of data science and big data analytics, where students are primarily taught data acquisition, cleaning, analytical and visualization skills. While these courses help students in developing skills related to transforming data into knowledge, it does not provide them with concepts and experiences related to hosting, storing, deploying and scaling up applications within performance and budgetary constraints. A few research-intensive universities offer specialized standalone courses [6, 7, 8, 9] such as “Cloud computing”, “Big data management” etc., where the abovementioned collective paradigm is addressed in a greater extent. However, being an optional course and offered at a handful of universities, only a small number of students receive the benefit. There is a big gap between the advances in big data and cloud computing and their inclusion in college-level instructions and this poster aims to address this gap.

We suggest an alternative approach to the abovementioned efforts that focuses on integrating topics of the collective paradigm as modules to the core courses. Each developed module spans two 75 minutes class periods and has specific learning goals, lessons plans, and assessment resources. A substantial advantage of the modular approach is that a large number of CS majors can be exposed to these contemporary topics and technologies via systematic and increasing integration throughout the computing curricula, and without the need of developing an additional core or elective course. In this poster, we report on our experiences in offering three such modules in respective courses and our objective is to share our experience with other instructors and to receive feedback about our approach. Interested readers are advised to explore resources posted at the project website [10].

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

SC'18, November 2018, Dallas, Texas USA

2. “Computer Architecture” Module

This module is integrated to the “Computer Architecture” class and is designed to expose students to virtualization and cloud computing fundamentals and to provide students hands-on experience in using AWS cloud.

Lesson Plan comprises discussion on important cloud computing characteristics such as scalability, on-demand access, measured services, and elasticity. The lecture then explains resource sharing and virtualization while pointing out the important aspects of it such as migration, time-sharing, isolation etc. The lesson plan also includes a tutorial session that instructs students the basic management of Amazon EC2 instances.

Assessment Instruments contain a hands-on project that involves exploring Amazon EC2 web services by creating various EC2 instances and by benchmarking these instances to determine how performance scales with the different virtual machine types. Students are provided with various benchmarking applications (CPU, IO) and are required to represent their results graphically and to report on them critically.

Overall, students were able to accomplish the project with 40% of them achieving A grade, 20% achieving B grade, 15% securing C grade, and 25% attaining D/F grades. A three questions survey was administered at the end of the module and student positively attested the enhancements due to the module. More specifically, 79% of the students found (strongly agreed or agreed) the conveyed topics interesting, 72% of them strongly agreed or agreed that they are well-versed on the topics, and 86% reported their desire to learn the topics more after the intervention.

3. “Analysis of Algorithm” Module

This module is designed to be included within an Algorithm class and the emphasis is on understanding the MapReduce programming framework, making use of Apache Hadoop, HDFS, MapReduce and Apache Spark in designing and deploying applications, and understanding the cost vs. performance tradeoff by executing analytics applications using various input data sizes and utilizing different execution setups in chameleon [11] cloud.

Lesson Plan includes lecture based on MapReduce Programming, Hadoop and HDFS, and Apache Spark Framework and a tutorial session based on running Hadoop through Cloudera’s VM [12] on the local machine and running Spark on Chameleon cloud environment.

Assessment Instruments include a hands-on project that involves modifying and developing MapReduce applications, and deploying Spark applications in the cloud while focusing on resource provisioning and performance vs. cost tradeoff.

All students were able to perform Task 1 and 2 (modifying MapReduce code to meet new requirements) of the Project. Task 3 (developing MapReduce program from scratch) was successfully completed by 85% of the students and Task 4 (running Spark application that analyzes Wikipedia pageview dataset [13] in the cloud and understand cost vs. performance) was completed by 30% of the students. Survey results show that all (100%) students found (strongly agreed or agreed) the conveyed topics interesting after the intervention. Similarly, 61% of the students strongly agreed or agreed that they are well-versed on the topics and 77% of the students expressed their eagerness to learn more.

4. “Database Management” Module

Currently, numerous application scenarios require processing very large datasets in a highly scalable and distributed fashion. Various types of big data systems have been designed to address this challenge and many of them have recognized the strengths of SQL as a query language. The proposed module for the introductory Database course is designed to expose students to the various types of big data systems and to integrate the study of SQL within those systems.

Lesson Plan includes lecture that discusses limitations of relational database systems, and key properties, strengths, limitations of various big data management system such as MapReduce, No-SQL, and New-SQL. The module also includes a tutorial session that provides students with hands-on experiences in Spark SQL.

Assessment Instruments include a hands-on project that involves developing Spark application that analyzes historical Facebook stock prices and use Spark SQL to query the data. In this project, students explored situations where the schema could be inferred from the data as well as the situation where it must be enforced programmatically.

Overall, students were able to accomplish Task 1 (infer schema automatically) while 63% of them attained A grade, 25% of them attained B grade and 12% attained C grade. Task 2 (define schema programmatically), however was successfully completed by only 12% of the students. 73% students attested (strongly agreed or agreed) the conveyed topics as interesting, 60% of the students strongly agreed or agreed that they are well-versed on the topics, and 87% of the students revealed their desire to learn the topics more.

ACKNOWLEDGMENTS

This research was supported by National Science Foundation Award # 1600864.

REFERENCES

- [1] Joint Task Force on Computing Curricula, Association for Computing Machinery (ACM) and IEEE Computer Society. 2013. Computer Science Curricula, 2013.

- [2] Criteria for accrediting computing programs, 2017-2018 ccreditation cycle, <http://www.abet.org/wp-content/uploads/2016/12/C001-17-18-CAC-Criteria-10-29-16-1.pdf>
- [3] B. Ramamurthy, "A Practical and Sustainable Model for Learning and Teaching Data Science", Proceedings of the 47th ACM Technical Symposium on Computing Science Education (SIGCSE), March, 2016.
- [4] S. J. Matthews, "Using Phoenix++ MapReduce to introduce undergraduate students to parallel computing", Journal of Computing Sciences in Colleges, vol. 32 Issue 6, June 2017.
- [5] J. Eckroth, "Teaching Future Big Data Analysts: Curriculum and Experience Report", IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), May, 2017.
- [6] A. S. Rabkin, C. Reiss, R. Katz, and D. Patterson. "Experiences teaching mapreduce in the cloud", In Proceedings of the 43rd ACM Technical Symposium on Computer Science Education (SIGCSE), 2012.
- [7] M. S. Rehman, J. Boles, M. Hammoud, M. F. Sakr, "A Cloud Computing Course: From Systems to Services, Proceedings of the 46th ACM Technical Symposium on Computer Science Education (SIGCSE), 2015.
- [8] CS5412: Cloud Computing, <http://www.cs.cornell.edu/courses/cs5412/2018sp/>
- [9] CS309: Cloud Computing <http://cs309a.stanford.edu/>
- [10] <http://ibigcloud.altl.org>, 2018.
- [11] Chameleon cloud, <https://www.chameleoncloud.org>
- [12] Cloudera, Cloudera QuickStart VM https://www.cloudera.com/downloads/quickstart_vms/5-12.html
- [13] Wikipedia Page Views <http://dumps.wikimedia.org/other/pagecounts-raw/>