Leveraging Computational Notebooks for Data Science MOOCs

April Y. Wang
School of Information
University of Michigan
Ann Arbor, MI, USA
aprilww@umich.edu

Steve Oney
School of Information
University of Michigan
Ann Arbor, MI, USA
soney@umich.edu

Christopher Brooks
School of Information
University of Michigan
Ann Arbor, MI, USA
brooksch@umich.edu

ABSTRACT

Computational notebooks enable data scientists to document their exploration process through a combination of code, narrative text, visualizations, and other rich media. In this position paper, we discuss the opportunities and challenges of adapting computational notebooks for data science MOOCs. In particular, we propose the redesign of computational notebooks from three dimensions: 1) supporting real-time group collaboration, 2) facilitating joint discourse over shared context, 3) encouraging active learning.

KEYWORDS

computational notebooks, data science education, MOOCs

INTRODUCTION

The rise of big data has increased the job demand for data scientists, which has been considered as the sexiest job of the 21st century [1]. Besides the growth of data science degree programs in colleges, the proliferation of data science MOOCs expands the access to quality education for learners at scale

Conference'17, July 2017, Washington, DC, USA
2019. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

who are seeking to build their skills in data science. Today, there are more than 100 data science related courses on Coursera, including a series of specialized training and online degree programs.

Many of these data science MOOCs have introduced Python programming in computational notebooks (e.g., Jupyter Notebook, Apache Zeppelin Notebook). In practice, Jupyter Notebook is the most popular tool for interactive data science [8]. As an application of exploratory programming [4], data scientists need to frequently inspect the outputs of parts of the code, which is well-supported by Jupyter Notebook. In addition, Jupyter Notebook allows users to document their exploration process using a combination of code, output, explanatory text, visualizations, and other media, which further enables sharing and reproducing the results. In data science MOOCs, instructors use computational notebooks to demonstrate code and its output. While learners navigate example notebooks to enhance their perceived knowledge from watching video lectures, or create their own notebooks for assignments or capstone projects.

In this paper, we reflect on the current practice of computational notebooks in online data science courses and describe our vision to leverage computational notebooks for data science MOOCs. In particular, we discuss opportunities and challenges for redesigning computational notebooks from three dimensions:

- Engaging learners through supporting real-time group collaboration in computational notebooks
- Facilitating joint discourse within computational notebooks
- Encouraging active learning through designing computational notebooks as a problem-based integrated learning environment

GROUP COLLABORATION IN NOTEBOOKS

Collaborative learning (CL) refers to groups of learners working together to solve a problem [2]. In general, collaborative learning is perceived useful for helping learners construct their understanding into a conceptual framework, internalizing knowledge through social discourse, and motivating learning. Data science is considered a practice of exploratory programming where the task is open-ended with no perfect solution [4]. Exploring possible solutions using different techniques and reasoning about the benefits and trade-offs are critical skills for data science learners to practice. While MOOCs offer learners the potential of interaction with learners at distance [how do distance learners connect], we believe that integrating group collaboration in data science MOOCs can help learners improve their understanding of the concepts through working together and communicating with peers. In addition, group collaboration can possibly improve MOOCs' completion rate by motivating and engaging learners.

However, one practical challenge for supporting collaborative learning in data science MOOCs is the lack of adequate tools. During rapid exploration, it can be burdensome for remote peers to share code snippets or data artifacts asynchronously. Although tools like Google Colaboratory allow multiple users editing the same notebook simultaneously, which helps maintain a shared understanding and reduce the communication cost, our observational study of real-time collaborative editing in Jupyter Notebook has revealed several challenges. For example, two users editing the same notebook may amplify the tension between exploration and keeping a clear explanation of the notebook, which is already identified as a problem in existing single-authored notebooks [9]. Thus, it would be worth exploring mechanisms such as encouraging planning and annotations, or retrieving related information from peer discussions to improve collaborative editing in Jupyter Notebook.

FACILITATING JOINT DISCOURSE

Discussion forums are one of the major support mechanisms for connecting remote learners with their peers and instructors asynchronously through joint discourse. The topics discussed by learners and instructors on data science MOOCs can be general (e.g., clarification of a concept) or problem based (e.g., discussions of alternative solutions for an assignment). However, the participation rate is generally perceived low on discussion forums. Learners who post the topic may fail to capture the problem context, while other learners who come across the post may hesitate to explore the problem due to the difficulties in setting up the environment, retrieving the same dataset, and running the intermediate code. In parallel, prior study has explored the benefit of sharing context through collaborative media curation for improving participation in online learning [3]. Therefore, we envision the design of a discussion forum based on computational notebooks. Learners can describe their questions within the notebook. Instructors and other learners can then reproduce the problem by executing the notebook, explore alternative solutions directly in the notebook, and attach comments to notebook cells.

ENCOURAGING ACTIVE LEARNING

The integrated approach that embeds comment threads, assessment [6] or interactive multimedia exercises [5] with lecture videos can improve the learning efficiency and better engage learners. Alternatively, we would like to explore the design of computational notebooks as an integrated learning environment. Instead of instructors recording their screen that demonstrates a computational notebook and learners navigating through lecture video, learners would navigate mainly through the computational narrative. The instructors' operations in the notebook would be recorded and played back to learners. Learners can pause the demonstration and edit the notebook at anytime they want. We would further examine how this mechanism can provide learners flexibility to learn at their own pace, and engage learners with active learning.

AUTHORS' BACKGROUND

The team has a general interest in exploring approaches and tools that effectively teaching programming on MOOCs. Previously, our team has studied ways to improve remote communication about code [7]. In addition, our team has participated in preparing and delivering an introductory Python programming course on campus and an applied data science course on the MOOC, where we asked learners to use computational notebooks. Through the teaching experience, our team gains empirical evidence of the challenges in adapting computational notebooks in data science education. We hope to contribute our understanding of computational notebooks and share our design insights with the human-centered data science community.

REFERENCES

[1] Thomas H. Davenport and D. J. Patil. 2012. Data Scientist: The Sexiest Job of the 21st Century. (2012). Issue October 2012. https://hbr.org/2012/10/data-scientist-the-sexiest-job-of-the-21st-century

[2] Jeanne Marcum Gerlach. 1994. Is this collaboration? 1994, 59 (1994), 5–14. https://doi.org/10.1002/tl.37219945903

[3] William A. Hamilton, Nic Lupfer, Nicolas Botello, Tyler Tesch, Alex Stacy, Jeremy Merrill, Blake Williford, Frank R. Bentley, and Andruid Kerne. 2018. Collaborative Live Media Curation: Shared Context for Participation in Online Learning. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, 555:1–555:14. https://doi.org/10.1145/3173574.3174129

[4] Mary Beth Kery and Brad A. Myers. 2017. Exploring exploratory programming. In *2017 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)* (2017-10). 25–29. https://doi.org/10.1109/VLHCC.2017.8103446

[5] Juho Kim, Elena L. Glassman, AndrÃľs Monroy-HernÃąndez, and Meredith Ringel Morris. 2015. RIMES: Embedding Interactive Multimedia Exercises in Lecture Videos. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI '15*. ACM Press, 1535–1544. https://doi.org/10.1145/2702123.2702186

[6] Toni-Jan Keith Palma Monserrat, Yawen Li, Shengdong Zhao, and Xiang Cao. 2014. L.IVE: An Integrated Interactive Video-based Learning Environment. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, 3399–3402. https://doi.org/10.1145/2556288.2557368

[7] Steve Oney, Christopher Brooks, and Paul Resnick. 2018. Creating Guided Code Explanations with Chat.Codes. *Proc. ACM Hum.-Comput. Interact.* 2, CSCW, Article 131 (Nov. 2018), 20 pages. https://doi.org/10.1145/3274400

[8] Jeffrey M. Perkel. 2018. Why Jupyter is data scientists' computational notebook of choice. *Nature* 563 (2018), 145. https://doi.org/10.1038/d41586-018-07196-1

[9] Adam Rule, Aurélien Tabard, and James D. Hollan. 2018. Exploration and Explanation in Computational Notebooks. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 32, 12 pages. https://doi.org/10.1145/3173574.3173606