Editorial

Factors to be considered in designing frameworks for automated bioinformatics pipelines—a perspective based on application setting

Yuan-Mao Hung[1], Liang-Chuan Lai[2,3]

[1]Graduate Institute of Biomedical Electronics and Bioinformatics, National Taiwan University, Taipei; [2]Bioinformatics and Biostatistics Core, Center of Genomic and Precision Medicine, National Taiwan University, Taipei; [3]Graduate Institute of Physiology, College of Medicine, National Taiwan University, Taipei

Correspondence to: Liang-Chuan Lai. Graduate Institute of Physiology, College of Medicine, National Taiwan University, Taipei. Email: llai@ntu.edu.tw.

Submitted Oct 29, 2020. Accepted for publication Nov 19, 2020.
doi: 10.21037/tcr-20-3168
View this article at: http://dx.doi.org/10.21037/tcr-20-3168

Most bioinformatics tools are developed in the command line interface. Although a command line interface is quite flexible for choosing the options of the software, it requires heavy typing work. Therefore, many software engineers prefer to develop automated versions of popular pipelines using a more convenient graphical user interface (GUI). However, a critical issue to be considered in the early development stage is the setting in which the automated tools will be applied, such as localhost, a local network, or the internet. In this article, we will discuss several factors to be considered regarding the application setting.

First, for use on the internet, integrating the automated pipeline with on-line platform is a good design strategy (1). On-line platform can largely reduce the installation time. When a developer releases the automated tool to the internet, people around the world can access it directly through browsers (2). This is a good opportunity to promote the developed software for commercial usage. The tools that can be used to develop an online platform are free, such as Django web-framework (3) for back-end design and html, CSS, and Javascript for front-end design, but several issues need to be considered, such as how to distribute computing resources, memory, and disk space across large numbers of users (4-6).

Second, if the tools are built to run on a local network, users can still easily access the tools through their browsers, but the software is accessible only to people in a more limited range (e.g., a building or an office). On the other hand, maintaining a tool in a local network is much simpler than releasing it to the internet. The developer only needs to consider the requirements of a limited number of users in a local area. Therefore, both computing resource distribution and security issues will be much simpler.

Third, if the tool can only work on a local machine, users will need to install the software on their own computer and resolve any error messages by themselves (7-9). In some cases, the computing resources of the personal computer may not be enough to handle large amounts of data, e.g., for genomic analyses. If the installation takes place on a multi-user workstation, the system may require each user to install the software in their home directory due to administrative issues. Therefore, the tool will be installed repeatedly by different users and waste disk space.

In summary, using online-platform design strategy to develop an internet-based system can save considerable disk space and installation time. A local network design is an appropriate framework for many automated pipelines, and makes it much easier to handle computing resource and security issues. Internet-based systems are also a good choice if there is not enough local computing power. These design framework problems need to be taken into consideration in the early development stage of any bioinformatics tool.

Acknowledgments

Funding: This work was supported by a grant from the Ministry of Science and Technology of Taiwan [MOST 109-2634-F-002-043].

Footnote

Provenance and Peer Review: This article was a free submission to the journal. The article did not undergo external peer review.

Conflicts of Interest: Both authors have completed the ICMJE uniform disclosure form (available at http://dx.doi.org/10.21037/tcr-20-3168). The authors have no conflicts of interest to declare.

Ethical Statement: The authors are accountable for all aspects of the work in ensuring that questions related to the accuracy or integrity of any part of the work are appropriately investigated and resolved.

References

1. Tolopka SJ, Light JJ. Secure and convenient information storage and retrieval method and apparatus. U.S. Patent No. 6,044,349, 2000.
2. Liang H, Huang C. Identification of tumor microenvironment-related genes in lower-grade gliomas by mining TCGA database. Transl Cancer Res 2020;9:4583-95.
3. Burch C. Django, a web framework using python: Tutorial presentation. J Comput Sci Coll 2010;25:154-5.
4. Afgan E, Baker D, Batut B, et al. The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2018 update. Nucleic Acids Res 2018;46:W537-44.
5. Huse SM, Mark Welch DB, Voorhis A, et al. VAMPS: a website for visualization and analysis of microbial population structures. BMC Bioinformatics 2014;15:41.
6. Quast C, Pruesse E, Yilmaz P, et al. The SILVA ribosomal RNA gene database project: improved data processing and web-based tools. Nucleic Acids Res 2012;41:D590-D596.
7. Bolyen E, Rideout JR, Dillon MR, et al. Reproducible, interactive, scalable and extensible microbiome data science using QIIME 2. Nat Biotechnol 2019;37:852-7.
8. Darling AE, Jospin G, Lowe E, et al. PhyloSift: phylogenetic analysis of genomes and metagenomes. PeerJ 2014;2:e243.
9. Li Y, Wang X, Shi L, et al. Predictions for high COL1A1 and COL10A1 expression resulting in a poor prognosis in esophageal squamous cell carcinoma by bioinformatics analyses. Transl Cancer Res 2020;9:85-94.

Cite this article as: Hung YM, Lai LC. Factors to be considered in designing frameworks for automated bioinformatics pipelines—a perspective based on application setting. Transl Cancer Res 2020;9(12):7382-7383. doi: 10.21037/tcr-20-3168

