

Comp Bio Helpers: Making Computational Biology Algorithms Accessible to Software Engineers

Kishitij Kaushal
kishitijk@vt.edu
Virginia Tech
Blacksburg, Virginia, USA

Eli Bullock-Papa
elibp13@vt.edu
Virginia Tech
Blacksburg, Virginia, USA

Connor Brodish
cbrodish03@vt.edu
Virginia Tech
Blacksburg, Virginia, USA

Ryan Zhang
ryanz22@vt.edu
Virginia Tech
Blacksburg, Virginia, USA

ABSTRACT

Many software engineers struggle to understand computational biology concepts, especially when dealing with outdated and poorly designed software tools. Our proposed solution is an easily accessible and highly interactive web application that combines various computational biology concepts, such as BLOSUM tables and sequence alignment algorithms, which can be used by both beginners and experts. By making these algorithms accessible and intuitive, we aim to bridge the gap between software engineers and computational biology.

KEYWORDS

computational biology, software engineering, algorithm visualization, bioinformatics, education, WebGPU, Next.js

ACM Reference Format:

Kishitij Kaushal, Connor Brodish, Eli Bullock-Papa, and Ryan Zhang. 2024. Comp Bio Helpers: Making Computational Biology Algorithms Accessible to Software Engineers. In . ACM, New York, NY, USA, 2 pages. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

1 INTRODUCTION

Computational biology is an interdisciplinary field that combines biology, computer science, and mathematics to analyze biological data using computational techniques. However, many software engineers find it challenging to learn about computational biology algorithms due to the complexity of existing tools, which are often outdated and difficult to use. Moreover, there is a lack of visualization tools to help users see how these algorithms work step-by-step, making it hard to understand the underlying principles. Our project, **Comp Bio Helpers**, aims to address these issues by offering an intuitive web-based platform that software engineers, students, and researchers can use to experiment with and visualize computational biology algorithms.

Permission to make digital or hard copies of all or part 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 components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
Conference'17, July 2017, Washington, DC, USA

© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM
<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

2 RELATED WORK

Several tools exist for computational biology, each serving different purposes and with unique strengths:

BLAST [1] is a commonly used algorithm that finds regions of local similarity between sequences and is effective for identifying sequence homologies and evolutionary relationships. Despite its power and efficiency, BLAST can be difficult to use for beginners because it lacks an intuitive visualization interface.

CLUSTAL W [3] is another popular tool used for multiple sequence alignment, improving the sensitivity of progressive alignments through sequence weighting and specific gap penalties. However, the tool's command-line nature makes it challenging for non-bioinformatics experts to use.

T-Coffee [2] offers fast and accurate multiple sequence alignment, providing flexibility in how sequences are weighted and aligned. T-Coffee, like other command-line tools, requires users to have a deep understanding of the algorithm's options, making it less accessible to software engineers without a background in computational biology.

These tools set a high standard for sequence alignment and visualization, but they require significant installation and command-line knowledge, which limits their accessibility to novice users. **Comp Bio Helpers** aims to address these challenges by providing a web-based platform that offers similar capabilities without the overhead, making computational biology algorithms more accessible to a broader audience.

3 PROPOSED SOLUTION

Our proposed solution is a web-based platform, **Comp Bio Helpers**, that integrates various computational biology algorithms and provides interactive visualizations to help users learn and experiment. The tool will include:

- Support for BLOSUM matrices, global alignment, and local alignment.
- Step-by-step visualizations of each algorithm's process.
- Real-time input handling for various sequence formats (e.g., space-delimited, newline-delimited).
- A highly performant front-end built using **Next.js** that leverages **WebGPU** for high-speed parallelized computations of alignment algorithms.

- An intuitive user interface that simplifies algorithm selection and comparison.

By using WebGPU for the front-end computational tasks, the application is able to offload complex alignment calculations to the client-side, making the platform faster and more responsive. This design choice not only reduces server load but also makes the tool usable even on devices with lower processing power.

4 SOFTWARE ENGINEERING PROCESS

We have chosen an agile development methodology, incorporating iterative design and regular stakeholder feedback. This approach allows us to rapidly prototype features and refine them based on user input. The project will be divided into three main phases:

- (1) **Requirements Gathering and Initial Design:** Collect user requirements using surveys and interviews, and create wireframes for the UI.
- (2) **Development and Testing:** Implement the core algorithms, UI, and visualizations using an iterative approach, focusing on optimizing WebGPU performance.
- (3) **Deployment and Maintenance:** Deploy the platform on a serverless infrastructure using modern cloud technologies, and implement monitoring and update mechanisms for on-going maintenance.

5 HIGH-LEVEL DESIGN AND IMPLEMENTATION

The platform will be divided into a front-end UI using Next.js, a WebGPU layer for algorithm computations, and a back-end server for

managing user settings and projects. The front-end will utilize React components, while the back-end will be developed with Node.js and express.js. Communication between the front-end and back-end will be handled via RESTful APIs. The system will be tested using a combination of unit testing for algorithms and integration testing for UI and server interactions.

6 CONCLUSION AND FUTURE WORK

Comp Bio Helpers fills a gap in computational biology education by providing software engineers and students with an accessible, web-based tool for learning complex algorithms. In the future, we plan to expand the platform to include additional computational biology algorithms and integrate machine learning techniques for advanced analyses. Limitations include computational load for large datasets and the need for real-time responsiveness for interactive elements.

7 REFERENCES

REFERENCES

- [1] Stephen F. Altschul, Warren Gish, Webb Miller, Eugene W. Myers, and David J. Lipman. 1990. Basic Local Alignment Search Tool. *Journal of Molecular Biology* 215, 3 (1990), 403–410. <https://blast.ncbi.nlm.nih.gov/>
- [2] Cedric Notredame, Des G. Higgins, and Jaap Heringa. 2000. T-Coffee: A novel method for fast and accurate multiple sequence alignment. *Journal of Molecular Biology* 302, 1 (2000), 205–217.
- [3] Julie D. Thompson, Des G. Higgins, and Toby J. Gibson. 1994. CLUSTAL W: Improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties, and weight matrix choice. *Nucleic acids research* 22, 22 (1994), 4673–4680.