PxBLAT: An Efficient and Ergonomics Python Binding Library for BLAT

Yangyang Li[1] **and Rendong Yang**[1,✉]

[1]Department of Urology, Northwestern University Feinberg School of Medicine, Chicago, IL 60611

PxBLAT provides an ergonomic and efficient Python binding library for BLAT, designed to enhance the user experience and performance of genomic analysis tasks. By providing a Pythonic interface to BLAT, PxBLAT simplifies the usage of BLAT, allowing incorporating its functionality directly into their Python-based bioinformatics workflows. Furthermore, A new parallelization and non-blocking model enables improved efficiency and intuitive user interface, thereby reducing the complexity of genomic data analysis tasks.

Software Libraries | Sequence Analysis | BLAT
Correspondence: ***rendong.yang@northwestern.edu***

Introduction

The continual advancement of genome sequencing technologies has led to an exponential increase in available genomic data. Tools to analyze and manipulate these data have become critically important in both research and clinical contexts. BLAST-like alignment tool (BLAT) (Kent, 2002) is a standout in the bioinformatics field for its capability to perform rapid genome sequence alignments. It offers a faster alternative to Basic Local Alignment Search Tool (BLAST) (Altschul et al., 1990) for aligning DNA sequences to the human genome (Kent, 2002). Despite its widespread use and acceptance in the bioinformatics community, interfacing with BLAT can present challenges, particularly when integrating it within a broader Python-based analytical pipeline. Since BLAT is implemented by C programming language and only provides a variety of utilities with a command line interface (CLI). So far, we lack a holistic approach to enhance both the performance and the usability of BLAT.

On the other hand, Python's rise as a favored programming language in bioinformatics is well-documented, due to its ease of use, extensive libraries, and versatility (Perkel, 2015). Various binding libraries have been developed to extend Python's reach into other computing languages, improving the flexibility and interoperability of bioinformatics tools. For instance, Biopython (Cock et al., 2009), one of the most prominent bioinformatics libraries, provides interfaces to tools like BLAST (Altschul et al., 1990), Clustal (Higgins and Sharp, 1988), and others. Nonetheless, to date, no comprehensive Python binding library for BLAT has been developed.

Here we propose PxBLAT, a modern Python library designed to streamline and enhance the interaction with BLAT, thereby making it more efficient and ergonomic. PxBLAT serves as a bridge, bringing the high-performance capabilities of BLAT into the Python environment, which is widely regarded for its readability, simplicity, and extensive library support. By improving the usability of BLAT and seamless integration within Python, PxBLAT opens up new possibilities for efficient genomic analysis. We provide evidence of its performance improvements, demonstrate its ergonomic advantages, and discuss its potential applications in genomic research. The overarching aim of this work is to fill the observed gap by providing a Python binding library specifically tailored for BLAT, addressing both its efficiency and ergonomic concerns.

Materials and Methods

This is materials and methods. Pybind11 (Jakob et al., 2016) blat (Kent, 2002)

Design Philosophy

The study aims to create an efficient, ergonomic Python binding library that enhances the interface and usage of the BLAT tool. We approached this by focusing on the crucial aspects of implementation design, library features, and integration.

The design of PxBLAT follows the Pythonic principles of readability and simplicity. It is built to be intuitive, reducing the learning curve for users familiar with Python and bioinformatics tools. In line with this, we focused on minimizing the complexity of the system while maximizing the usability and performance of BLAT.

PxBLAT is implemented in Python, employing ctypes, a Python library, to interface with the BLAT's C code. This approach enabled direct access to the functions of the BLAT program without modifying its original source code, preserving the integrity and performance of BLAT while extending its capabilities.

PxBLAT includes a range of features to enhance the user experience with BLAT: Seamless Integration: PxBLAT is designed to work cohesively within the Python environment, allowing the user to utilize the BLAT functions as if they were Python functions. It is compatible with popular Python bioinformatics libraries like Biopython and NumPy. Efficiency: By accessing BLAT's functionality directly from Python, PxBLAT re-

duces the need for system calls, improving the overall efficiency of bioinformatics pipelines that use BLAT. Error Handling: PxBLAT incorporates robust error handling, ensuring that BLAT-related errors are captured and communicated to the user in a Pythonic manner. Documentation and Examples: PxBLAT comes with comprehensive documentation and a suite of example scripts to demonstrate how to use the library effectively. In conclusion, the design and implementation of PxBLAT align with our goal of providing an efficient and ergonomic Python binding library for BLAT. The developed library unlocks new possibilities for efficient genomic analysis, especially within the context of Python-based workflows.

Implementation

Type hints are provided for all public classes and functions, allowing a static analyzer such as MyPy (https://mypy-lang. org) to detect type errors ahead of runtime. These type annotations also make PyHMMER more pleasant to use inside an Integrated Development Environment (IDE), where the function signatures can be suggested and corrected automatically.

Results

PxBLAT has consistent result with BLAT

Benchmarking Performance

Ergonomics

It is possible to add a one-column Figure like this (Figure 2).

Discussion

This is the discussion section where you wax lyrical about your findings.

Acknowledgements

Conflict of interest

Funding

Data availability

The code is available is GitHub PxBLAT. The benchmarking dataset and code is located in GitHub as well.

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. Basic local alignment search tool. *Journal of molecular biology*, 215(3):403–410, 1990.

Cock, P. J., Antao, T., Chang, J. T., Chapman, B. A., Cox, C. J., Dalke, A., Friedberg, I., Hamelryck, T., Kauff, F., Wilczynski, B., et al. Biopython: freely available python tools for computational molecular biology and bioinformatics. *Bioinformatics*, 25(11):1422–1423, 2009.

Higgins, D. G. and Sharp, P. M. Clustal: a package for performing multiple sequence alignment on a microcomputer. *Gene*, 73(1):237–244, 1988.

Jakob, W., Rhinelander, J., and Moldovan, D. pybind11 — seamless operability between c++11 and python, 2016. https://github.com/pybind/pybind11.

Kent, W. J. Blat—the blast-like alignment tool. *Genome research*, 12(4):656–664, 2002.

Perkel, J. M. Programming: Pick up python. *Nature*, 518(7537):125–126, 2015.

Figure 1. These are cells.
(**A**) This is a regular figure with a legend as a caption underneath. Inset: 3X zoom. Scale bar, 10 µm.

Figure 2. This is a nucleus.
(**A**) This is a one-column figure with a legend as a caption underneath.

Supplementary Information

Figure S1. This is an endosome.
(**A**) This is a supplementary figure shown as a two-column image with a legend underneath.