BuddySuite: Command-line toolkits for manipulating sequences, alignments, and phylogenetic trees

Stephen R. Bond, Karl E. Keat, Sofia N. Barreira, and Andreas D. Baxevanis*
Computational and Statistical Genomics Branch, Division of Intramural Research, National Human Genome Research Institute, National Institutes of Health, 50 South Drive, Bethesda, MD, USA, 20892
***Corresponding author:** E-mail: andy@mail.nih.gov
Associate Editor:

Abstract

The ability to manipulate sequence, alignment, and phylogenetic tree files has become an increasingly important skill in the life sciences, whether to generate summary information or to prepare data for further downstream analysis. The command line can be an extremely powerful environment for interacting with these resources, but only if the user has the appropriate general-purpose tools on hand. BuddySuite is a collection of four independent yet interrelated command-line toolkits that facilitate each step in the workflow of sequence discovery, curation, alignment, and phylogenetic reconstruction. Most common sequence, alignment, and tree file formats are automatically detected and parsed, and over 100 tools have been implemented for manipulating these data. The project has been engineered to easily accommodate the addition of new tools, it is written in the popular programming language Python, and is hosted on the Python Package Index and GitHub to maximize accessibility. Documentation for each BuddySuite tool, including usage examples, is available at http://tiny.cc/buddysuite_wiki. All software is open source and freely available through http://research.nhgri.nih.gov/software/BuddySuite

Key words: software, command line, sequence, alignment, phylogenetic tree, Python

Letter

Introduction

Manipulation of biological sequence data is now a routine task within the life sciences, performed not just by bioinformaticians but also by 'bench biologists' who are becoming increasingly savvy in applying computational methods to their own work. While there are excellent graphical platforms for organizing, visualizing, and manipulating various (and often disparate) forms of data, it can be advantageous to interact with text files directly from the command line. Common tasks may include searching for specific records in a file, extracting subsequences, converting between formats, identifying motifs, or stripping poorly aligned regions from a multiple sequence alignment. While all of these can be accomplished with standard UNIX commands or existing open source software, combining tasks into a workflow may require the user to create a series of intermediate files, write complicated command-line operations, and move data between standalone tools or online services.

As an alternative we present BuddySuite, a comprehensive set of general-purpose command-line tools for manipulating sequence, alignment, and phylogenetic tree data that can be joined into reproducible workflows using a simple unified syntax.

The European Molecular Biology Open Software Suite (EMBOSS) (Rice *et al.*, 2000) and Biopieces are the most comprehensive general-purpose open-source bioinformatics toolkits currently available for the command line. While both are excellent software packages, BuddySuite includes a number of new features we believe will benefit biologists. In particular is our switch away from the 'one program per function' paradigm that EMBOSS and Biopieces employ. BuddySuite groups all functions related to a particular data type together into specific modules and uses flags to differentiate among them; this ensures a unified and predictable user interface and occupies a much smaller namespace on a user's system (EMBOSS and Biopieces each contains about 200 separate programs). Furthermore, file format detection is fully automated; any number of sequence, alignment, or phylogenetic tree files can be passed into their respective BuddySuite program, in any combination of supported formats, and the records will be parsed seamlessly (see table 1 for a list of supported formats). This is particularly useful when using the BuddySuite modules to call third party alignment or phylogenetic inference programs, as any idiosyncratic format conversions are handled without the need of additional input from the user.

Table 1. File format support for reading (R) and writing (W) provided by each BuddySuite module.

Format	SeqBuddy	AlignBuddy	PhyloBuddy
Clustal	R & W†	R & W	None
EMBL‡	R & W	R†/ W	None
FASTA	R & W	R†/ W	None
GenBank‡	R & W	R†/ W	None
Nexus	R & W†	R & W	R & W
Newick	None	None	R & W
NeXML	None	None	R & W
PHYLIP (interleaved)	R & W†	R & W	None
PHYLIP (sequential)	R & W†	R & W	None
SeqXML	R & W	None	None
Stockholm	R & W†	R & W	None
SWISS-PROT‡	R only	None	None

†All sequences must be the same length
‡Supports rich sequence annotation

One of the greatest advantages BuddySuite has over other tools is its handling of sequence feature annotation. Rich flat file formats like GenBank and EMBL support annotation, but this information is generally discarded by the EMBOSS programs and Biopieces is unable to generate these formats. BuddySuite modules are aware of features in the sequence records they process and will update those annotations when sequences are modified. For example, if a GenBank cDNA file is translated to protein, the relative positions of each feature will be scaled by one third to account for the conversion of codons to amino acids. Similarly, if those protein sequences are passed into a supported multiple sequence alignment program, such as MAFFT, the feature positions will be adjusted to account for gaps.

Command-line user interface

The four core command-line programs distributed with BuddySuite are SeqBuddy, AlignBuddy, PhyloBuddy, and DatabaseBuddy. The first three accept sequence, alignment, or phylogenetic tree data as input, respectively, using flags to switch among the tools available in each program. All output is printed directly to the terminal window by default and each module adheres to the UNIX convention of accepting piped data, allowing individual tools to be 'daisy-chained' into more complex workflows (illustrated further in the 'Use-case examples' section below). DatabaseBuddy, on the other hand, is intended to run primarily as a 'live shell', allowing the user to interactively search and download sequence data stored in the NCBI, UniProt, and Ensembl public databases. The current stable release version of BuddySuite (V1.2) includes 104 individual command-line tools across the four programs.

Use-case examples

BuddySuite modules are executed from the command line using the following generalized syntax:

```
$: module file(s) <cmd> <args> <modifiers>
```

Any number of files may be passed into the module but only a single command can be executed at a time. As a specific example, the following would accept two sequence files (in FASTA and GenBank formats) and delete any sequences larger than 300 residues (module names have been shortened in the following examples to sb, alb, and pb for SeqBuddy, AlignBuddy, and PhyloBuddy, respectively):

```
$: sb seqs1.gb seqs2.fa --delete_large 300
```

Whichever file format is encountered last will also be the output format of the final records (in this case, FASTA), although this behaviour may be overridden with the '--output' modifier:

```
$: sb seqs1.gb seqs2.fa --delete_large 300
   --output genbank
```

Modifiers like '--output' are used sparingly in the BuddySuite modules and only when their effects are intuitively applicable across most tools in the module (e.g., '--quiet' execution or to rewrite files '--in_place').

Complex workflows can also be built with the BuddySuite modules using the pipe character. In the following example, transmembrane domains (TMD) are identified in a set of homologous cDNAs (note that SeqBuddy recasts the format to GenBank when applying new sequence features), sequences with less than four TMDs are removed, a multiple sequence alignment is generated from the remaining sequences, the region of the alignment between the first and second conserved TMD is extracted (determined by looking at the alignment in GenBank format), and then phylogenetic inference software is called via PhyloBuddy to generate a tree which is then rooted at its midpoint.

```
$: sb  sequences.fa --transmembrane_domains |
   sb  --pull_records "TMD4" |
```

```
sb  --translate |
alb --generate_alignment mafft |
sb  --extract_regions "13:92" |
pb  --generate_tree raxmlHPC-SSE3 |
pb  --root
```

To piece this workflow together without BuddySuite, a user may use a command-line version of software like TMHMM (Krogh *et al.*, 2001) or access a web service like TOPCONS (Tsirigos *et al.*, 2015) to identify the TMDs, then parse the output file with awk or excel to create a list of sequence IDs with the correct number of TMDs. After pulling those sequences from the original file and translating them with seqret and transeq (from EMBOSS), they would need to be saved as a FASTA file to be passed into MAFFT to generate a multiple sequence alignment. Manual inspection or a custom script would be required to match the location of each TMD to the alignment, and then seqret would be called once again to extract the correct regions and create a new alignment file. RAxML could then be used to infer a phylogenetic tree and root it.

While the above pipeline is valid, BuddySuite offers a solution with fewer steps, fewer intermediary files, less manual intervention, and a consistent syntax.

Installation

The BuddySuite libraries have been written in Python 3 for use on all major operating systems (Windows 7+, Mac OSX, and Linux). Stable release versions of BuddySuite can be installed directly from the Python Package Index (PyPI) using the popular package manager 'pip', and the most recent development version is continually available from GitHub. While optional, users are also encouraged to run the BuddySuite configuration script after installation:

```
$: pip install buddysuite
$: buddysuite -setup
```

Doing so will create directories for caching data on the user's system and will register an email address for the tools that interact with public databases over an internet connection (to prevent possible IP blocking). To simplify installation, dependencies have been limited to packages available through PyPI, although a number of optional third-party programs can be accessed through BuddySuite as well; these include BLAST for comparing sequences, multiple sequence alignment packages like MAFFT, and phylogenetic inference packages like RAxML. As these programs are not necessary for the general operation of the BuddySuite modules, installation is at the user's descretion. The third-party tools that BuddySuite can interact with are itemized in table 2.

Documentation

Basic help is available for each BuddySuite module from the command line by passing in the '-h' or '--help' flag. Doing so will generate the list of available utilities along with brief usage instructions. Extended documentation is available as a public wiki (http://tiny.cc/buddysuite_wiki),

Table 2. List of optional third party software that BuddySuite programs can interact with.

	Program	Reference
SeqBuddy	BLAST	(Camacho et al., 2009)
AlignBuddy	ClustalΩ	(Sievers et al., 2011)
	ClustalW2	(Larkin et al., 2007)
	MAFFT	(Katoh and Standley, 2013)
	MUSCLE	(Edgar, 2004)
	PAGAN	(Löytynoja et al., 2012)
	PRANK	(Löytynoja and Goldman, 2005)
PhyloBuddy	FastTree	(Price et al., 2010)
	RAxML	(Stamatakis, 2006)
	PhyML	(Guindon et al., 2010)

BuddySuite performs all necessary format conversion to call any of these tools and, where appropriate, returns the result in the same format as the input. This is particularly useful when creating multiple sequence alignments from annotated sequences in GenBank or EMBL format.

complete with an explanation for any arguments and fully worked examples.

Developers

Looking forward, the modular nature of BuddySuite makes it particularly well suited to open-ended development. New tools are easily added to each existing module and new modules may eventually extend the suite to new data types. While we will continue to support and expand BuddySuite ourselves, we also strive to attract contribution from the broader community. To minimize barriers against community-driven development, the project is maintained on GitHub, has comprehensive unit test coverage of over 95%, includes extensive and accessible documentation, and makes every effort to conform with open-source best practices (Leprevost *et al.*, 2014; Seemann, 2013).

Instead of relying exclusively on active input from users to inform future direction, we have implemented an optional passive data collection system to monitor usage and report crashes. If a user chooses to opt into this software improvement program during installation, anonymized data will be periodically transmitted to an FTP server, allowing us to better tailor our efforts to the needs of the community. Furthermore, if a crash occurs that relates to a bug which is patched by a newer release, the software improvement monitoring system will inform the user about the update.

The internal functionality of BuddySuite is also easily accessed by developers wishing to write third-party Python programs. Each module has a core 'Buddy' class that automatically processes a variety of input types (including plain text, file paths, file handles, and lists of record objects), performs all necessary file format processing, and exposes methods for managing and writing the sequence or tree records. The functions in each library accept these 'Buddy' objects as input and generally return them as output, thus providing a standardized application programing interface that facilitates interoperability among functions. Once installed, the BuddySuite libraries can be imported using conventional Python syntax.

Conclusions

BuddySuite has been designed from the ground up as an intuitive, extensible, and unified platform for routine command-line tasks performed on sequence, alignment, and phylogenetic tree files. This is the first time such a large suite of general-purpose bioinformatics utilities have been

implemented purely in Python and packaged together under a flag-driven paradigm. Well-designed and actively supported open-source tools will be invaluable over the coming years as an increasing number of biologists turn to the command line to analyze their data. We hope that BuddySuite will be widely adopted by the community and, thanks to the passive data-collection features built into this project, we look forward to tailoring future development to the needs of our users.

Acknowledgments

This research was supported by the Intramural Research Program of the National Human Genome Research Institute, National Institutes of Health. We would also like to thank Drs. Maxence LeVasseur and Tyra Wolfsberg for their thoughtful feedback on this manuscript and the community members who contributed code to the project, big or small. It takes a village.

References

Camacho, C., Coulouris, G., Avagyan, V., Ma, N., Papadopoulos, J., Bealer, K., and Madden, T. L. 2009. BLAST+: architecture and applications. *BMC bioinformatics*, 10: 421.

Edgar, R. C. 2004. MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic acids research*, 32(5): 1792–1797.

Guindon, S., Dufayard, J.-F., Lefort, V., Anisimova, M., Hordijk, W., and Gascuel, O. 2010. New algorithms and methods to estimate maximum-likelihood phylogenies: assessing the performance of PhyML 3.0. *Systematic biology*, 59(3): 307–321.

Katoh, K. and Standley, D. M. 2013. MAFFT multiple sequence alignment software version 7: improvements in performance and usability. *Molecular biology and evolution*, 30(4): 772–780.

Krogh, A., Larsson, B., von Heijne, G., and Sonnhammer, E. L. L. 2001. Predicting transmembrane protein topology with a hidden markov model: application to complete genomes. *Journal of molecular biology*, 305(3): 567–580.

Larkin, M. A., Blackshields, G., Brown, N. P., Chenna, R., McGettigan, P. A., McWilliam, H., Valentin, F., Wallace, I. M., Wilm, A., Lopez, R., Thompson, J. D., Gibson, T. J., and Higgins, D. G. 2007. Clustal W and Clustal X version 2.0. *Bioinformatics*, 23(21): 2947–2948.

Leprevost, F. d. V., Barbosa, V. C., Francisco, E. L., Perez-Riverol, Y., and Carvalho, P. C. 2014. On best practices in the development of bioinformatics software. *Frontiers in genetics*, 5: 199.

Löytynoja, A. and Goldman, N. 2005. An algorithm for progressive multiple alignment of sequences with insertions. *Proceedings of the National Academy of Sciences of the United States of America*, 102(30): 10557–10562.

Löytynoja, A., Vilella, A. J., and Goldman, N. 2012. Accurate extension of multiple sequence alignments using a phylogeny-aware graph algorithm. *Bioinformatics*, 28(13): 1684–1691.

Price, M. N., Dehal, P. S., and Arkin, A. P. 2010. FastTree 2–approximately maximum-likelihood trees for large alignments. *PloS one*, 5(3): e9490.

Rice, P., Longden, I., and Bleasby, A. 2000. EMBOSS: the European Molecular Biology Open Software Suite. *Trends in genetics*, 16(6): 276–277.

Seemann, T. 2013. Ten recommendations for creating usable bioinformatics command line software. *GigaScience*, 2(1): 15.

Sievers, F., Wilm, A., Dineen, D., Gibson, T. J., Karplus, K., Li, W., Lopez, R., McWilliam, H., Remmert, M.,

Söding, J., Thompson, J. D., and Higgins, D. G. 2011. Fast, scalable generation of high-quality protein multiple sequence alignments using Clustal Omega. *Molecular systems biology*, 7(1): 539–539.

Stamatakis, A. 2006. RAxML-VI-HPC: maximum likelihood-based phylogenetic analyses with thousands of taxa and mixed models. *Bioinformatics*, 22(21): 2688–2690.

Tsirigos, K. D., Peters, C., Shu, N., Käll, L., and Elofsson, A. 2015. The TOPCONS web server for consensus prediction of membrane protein topology and signal peptides. *Nucleic acids research*, 43(W1): W401–7.