NAISL
Quarterly, 2017
Volume 1, Number 1
Pages 59 – 64
Print ISSN: 2588-6401
Online ISSN: 2588-641X

Review and compare Python and Perl programming languages used in bioinformatics

Armin Ahmadinasab[1*] and Navid Ahmadinasab[2]

Abstract

Bioython and Bioperl are open source tools, with the aim of providing libraries have been developed to solve problems in bioinformatics and life sciences. Two high-level languages Python and Perl are widely used in science, education and business. One problems of researchers in laboratory and bioinformatics methods, choice of the programming language to perform computer simulations for biological systems. Considering the genomic sequence analysis, the prediction of three-dimensional protein structure, functional analysis at the genome level, the establishment and management of databases and mathematical modeling and life processes in the bioinformatics laboratories, The suitableprogramming language can have a great impact on improving the quality of the output, reducing the time and memory required. In this paper, Python and Perl programming languages are beign compare to determine the time used in Windows and Linux operating systems and memory required to run standard bioinformatics algorithms, have been compared. The global alignment algorithm and neighbor-Joining are used for both Python and Perl. In this study, Perl is better than Python for I/O operations in terms of time and memory usage. But according to the global alignment algorithm and neighbor-Joining programs' results, although Python have better character string manipulation abilities, nevertheless Perl is better than Python for parsing a BLAST file.

Key Words

**Biopython,
Bioperl,
Bioinfomatics,
Life Siences,
Computational biology**

(∗)Corresponding author.
1. Department of Industrial Engineering, K.N. Toosi University of Technology, Tehran, P.O. Box 15875-4416, Iran. E-mail: armin4394@gmail.com, Phone Number: 09171574458
2. Department of Nanotechnology and Advanced Materials, Materials and Energy Research Center (MERC), Tehran, P.O. Box 31787-316, Iran. E-mail: na.84ir@gmail.com, Phone Number: 09126018946

فصلنامه علمی
سال اول، شماره ۱
صفحات ۵۹ – ۶۴، ۱۳۹۶
شاپای چاپی: ۶۴۰۱-۲۵۸۸
شاپای الکترونیکی: ۶۴۱X-۲۵۸۸

بررسی و مقایسه زبان‌های برنامه‌نویسی پایتون و پرل مورد استفاده در بیوانفورماتیک

آرمین احمدی نسب[1*] و نوید احمدی نسب[2]

چکیده

بیوپایتون و بیوپرل دو ابزار متن باز، با هدف ارائه کتابخانه‌های وسیع برای حل مسائل بیوانفورماتیک توسعه یافته‌اند. دو زبان سطح بالای پایتون و پرل به طورگسترده در زمینه‌های علمی، آموزشی و تجاری مورد استفاده قرار می‌گیرند. یکی از مشکلات محققان در روش‌های آزمایشگاهی و بیوانفورماتیک، انتخاب زبان برنامه‌نویسی مناسب جهت انجام شبیه‌سازی رایانه‌ای برای سامانه‌های زیستی است. با توجه به انجام فعالیت‌های تحلیل توالی‌های ژنوم، پیش‌بینی ساختار سه بعدی پروتئین‌ها، تحلیل کارکردی در سطح ژنوم، ایجاد و مدیریت پایگاه‌های داده‌ای و مدل‌سازی ریاضی و فرآیندهای حیات در آزمایشگاه‌های بیوانفورماتیک به صورت درون رایانه‌ای، انتخاب زبان برنامه‌نویسی مناسب، می‌تواند بر روی بالابردن کیفیت خروجی، کاهش زمان و حافظه مورد نیاز، تأثیر به سزایی داشته باشد. در این مقاله، دو زبان برنامه‌نویسی پایتون و پرل، با هدف تعیین زمان مصرفی در سیستم‌عامل‌های ویندوز و لینوکس و حافظه مورد نیاز جهت اجرای الگوریتم‌های بیوانفورماتیک استاندارد، با هم مقایسه شده‌اند. دو الگوریتم هم‌ترازی سراسری و اتصال همسایگی برای بررسی دو زبان پایتون و پرل مورد استفاده قرارگرفته است. بررسی‌ها نشان دهنده این موضوع است که در انجام عملیات ورودی/خروجی پرل از نظر زمان و حافظه مصرفی عملکرد بهتری نسبت به پایتون دارد. اما با توجه به نتایج بدست آمده از برنامه‌های هم‌ترازی سراسری و اتصال همسایگی، پایتون نسبت به پرل از کارایی بالاتری برای اعمال تغییرات بر روی کاراکترهای رشته‌ای برخوردار است. با این وجود، پایتون نسبت به پرل عملکرد ضعیف‌تری را برای تجزیه یک فایل BLAST داشته است.

آرمین احمدی‌نسب

نوید احمدی‌نسب

واژگان کلیدی

بیوپایتون،
بیوپرل،
بیوانفورماتیک،
علوم زیستی،
محاسبات زیست شناسی

(*) مسئول مکاتبات.
۱. دانشکده صنایع، دانشگاه صنعتی خواجه نصرالدین طوسی، آدرس ایمیل: armin4394@gmail.com،
تلفن: ۰۹۱۷۱۵۷۴۴۵۸
۲. پژوهشکده فناوری نانو و مواد پیشرفته پژوهشگاه مواد و انرژی، na.84ir@gmail.com،
تلفن: ۰۹۱۲۶۰۱۸۹۴۶

۱ مقدمه

بیوپایتون [۱] و بیوپرل [۲] دو ابزار متن باز و قابل دسترس برای همه سیستم عامل‌های اصلی از جمله ویندوز، لینوکس و مکینتاش هستند. دو زبان سطح بالای پایتون و پرل به طور گسترده در زمینه‌های علمی، آموزشی و تجاری مورد استفاده قرار می‌گیرند. از ویژگی‌های این دو زبان می‌توان به دستورات نحوی ساده برای یادگیری، ظرفیت‌های برنامه‌نویسی شیگرا [۳] و دارای اطلاعات کتابخانه‌ای زیاد اشاره کرد. پایتون و پرل زبان‌های اسکریپت‌نویسی نام‌گذاری می‌شوند و در هنگام اجرا، بدون ایجاد یک فایل واسط، به نمایش درآمده و مورد تفسیر قرار می‌گیرند [۶]. هر دو زبان از مدیریت حافظه خودکار و کتابخانه‌های بزرگ استفاده می‌کنند و مناسب برای برنامه‌نویسی وب و پیاده‌سازی تجزیه کننده مانند پایگاه داده اینترپرو [۴] هستند [۱۴].

با توجه به انجام فعالیت‌های تحلیل توالی‌های ژنوم، پیش‌بینی ساختار سه بعدی پروتئین‌ها، تحلیل کارکردی در سطح ژنوم، ایجاد و مدیریت پایگاه‌های داده‌ای و مدل‌سازی ریاضی و فرآیندهای حیات در آزمایشگاه‌های بیوانفورماتیک به صورت درون رایانه‌ای، انتخاب زبان برنامه‌نویسی مناسب، می تواند بر روی بالابردن کیفیت خروجی، کاهش زمان و حافظه مورد نیاز، تأثیر به سزایی داشته باشد.

هدف از این مقاله، انجام مقایسه‌ای بین دو زبان برنامه‌نویسی پایتون و پرل است. در ابتدا به معرفی و بیان ویژگی‌های اصلی دو زبان برنامه‌نویسی پایتون و پرل پرداخته‌ایم و در انتها، از نظر زمان مصرفی در دو سیستم عامل ویندوز و لینوکس به طور جداگانه و همچنین میزان حافظه مورد نیاز جهت اجرای الگوریتم‌های هم‌ترازی سراسری [۵] و اتصال همسایگی [۶] تحقیق به عمل آمده است.

۲ معرفی زبان‌های برنامه‌نویسی پایتون و پرل

۱.۲ پایتون

پایتون یک زبان برنامه‌نویسی شیگرا، تفسیر شده [۷] و انعطاف‌پذیر است که در انجام محاسبات علمی، پردازش فایل‌های متنی و وظایف رایج خودکار مورد استفاده قرار می‌گیرد [۳، ۴]. به طور اساسی، هدف از بیوپایتون، امکان استفاده از پایتون برای بیوانفورماتیک به وسیله ایجاد کلاس‌ها و افزونه‌های با قابلیت استفاده مجدد [۸] و کیفیت بالا است. قابلیت‌های اصلی بیوپایتون عبارت است از:

۱. توانایی تجزیه فایل‌های بیوانفورماتیک به ساختمان داده‌های قابل قبول در پایتون (جدول ۱).

۲. فایل‌ها در فرمت‌های پشتیبان شده می‌توانند رکورد به رکورد تکرار شوند یا از طریق رابط واژه‌نامه [۹] اندیس‌گذاری یا قابل دسترس باشند.

۳. معامله کد با وب سایت‌های معروف بیوانفورماتیک مانند NCBI و ExPASy.

۴. ارتباط با برنامه‌های رایج بیوانفورماتیک مانند BLAST، Clustalw و EMBOSS.

۵. یک کلاس توالی استاندارد که به توالی‌ها، شناسه توالی‌ها و ویژگی‌های آن‌ها می‌پردازد.

۶. ابزاری برای اجرای عملیات رایج روی توالی‌ها.

۷. کد برای اجرای طبقه‌بندی داده‌ها با استفاده از الگوریتم k نزدیک ترین همسایه‌ها [۱۰]، بیز ساده [۱۱] یا ماشین‌های بردار پشتیبان [۱۲].

---

[۱] Biopython
[۲] Bioperl
[۳] Object Oriented
[۴] InterPro
[۵] Global alignment
[۶] Neighbor-Joining
[۷] Compiled
[۸] Reusable
[۹] Dictionary
[۱۰] k-Nearest Neighbors
[۱۱] Naive Bayes
[۱۲] Support Vector Machines

۸. آسان بودن تقسیم وظایف با قابلیت موازی‌سازی به فرآیندهای جداگانه توسط کد.

۹. برنامه‌های مبتنی بر رابط کاربری گرافیکی برای اعمال تغییرات پایه‌رو توالی‌ها، ترجمه و ...

۱۰. یکپارچه بودن با بیواس‌کیوال [۱] [۲].

جدول ۱: فرمت‌های پشتیبان شده توسط پایتون

فرمت	خواندن	نوشتن	نام
.fasta	*	*	FASTA [۱۰]
.genbank	*	*	GenBank [۲]
.embl		*	EMBL [۸]
.swiss		*	Swiss-Prot/TrEMBL [۱۳]
.clustal	*	*	ClustalW [۱۲]
.phylip	*	*	PHYLIP [۵]
.stockholm	*	*	Pfam or Stockholm [۱]
.nexus	*	*	NEXUS [۹]

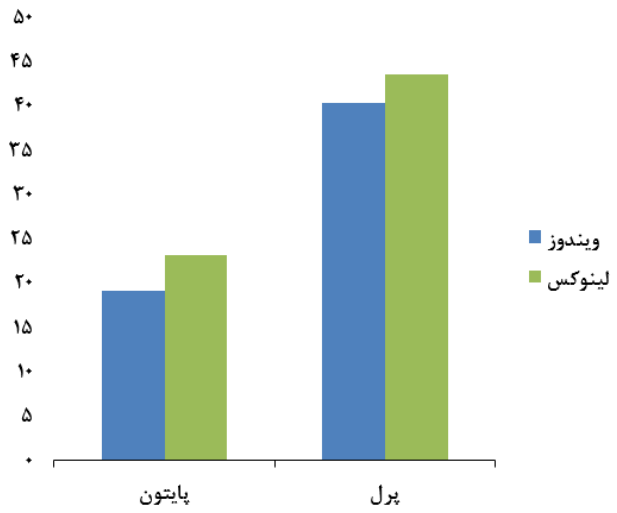
۲.۲ پرل

بیوپرل یک پروژه متن‌باز از مجموعه جامع کتابخانه‌های توسعه یافته به وسیله زبان برنامه‌نویسی پرل برای مدیریت و اعمال تغییرات بر روی اطلاعات علوم زیستی است. پرل در برقراری ارتباط با برنامه‌های کاربردی به منظور تحلیل توالی، تبدیل فرمت فایل‌ها، و استخراج اطلاعات از خروجی برنامه‌های تحلیلی و دیگر فایل‌های متنی بسیار موفق عمل کرده است [۱۱]. پرل می‌تواند با اطلاعات در فایل‌های متنی اسکی [۲] یا تخت که دقیقا فرمت‌های مهم داده‌های زیست‌شناسی در پایگاه داده‌های GenBank و PDB هستند کار کند. پرل، پردازش و اعمال تغییرات روی توالی‌های طولانی مانند DNA و پروتئین را آسان می‌کند. یکی از ویژگی‌های مهم استفاده از پرل و علت اصلی انتخاب آن برای پیشبرد تحقیقات علوم زیستی و نمونه‌سازی سریع توسط برنامه‌نویس است.

قابلیت حمل به معنی این است که چه تعداد از زبان‌های سیستم‌های کامپیوتری می‌توانند آن را اجرا کنند. پرل، با دارا بودن ویژگی قابلیت حمل بسیار بالا، برروی تمامی کامپیوترهای مدرن در آزمایشگاه‌های زیستی قابل دسترسی است. پرل سرعت بالایی در اجرا کردن برنامه‌ها دارد، ولی بهترین انتخاب نیست. برای بالابردن سرعت اجرا، رایج‌ترین زبان برنامه‌نویسی برای انتخاب، زبان C است [۷].

۳ مقایسه بین پایتون و پرل

در شکل ۱ سرعت الگوریتم هم‌ترازی سراسری توالی نوشته شده با دو زبان برنامه‌نویسی پرل و پایتون مقایسه شده است برنامه در سیستم‌عامل‌های لینوکس و ویندوز اجرا شده است.

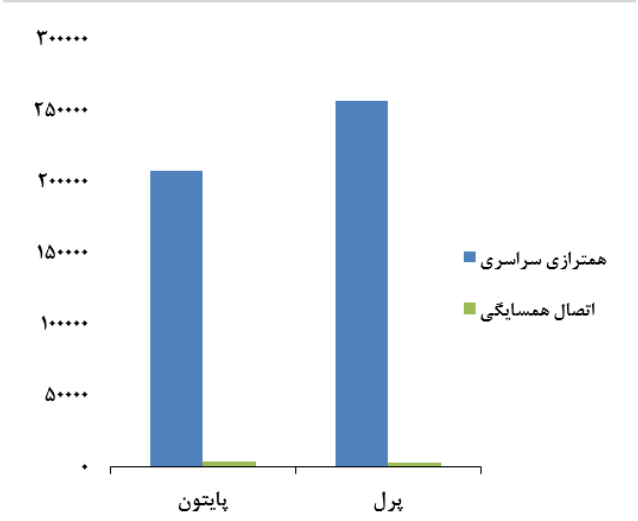


شکل ۱: مقایسه سرعت الگوریتم هم‌ترازی سراسری توالی نوشته شده با دو زبان برنامه‌نویسی پرل و پایتون. (از دو توالی DNA با ۳۲۱۶ جفت باز و ۳۲۱۷ جفت باز استفاده شده است) [۶].

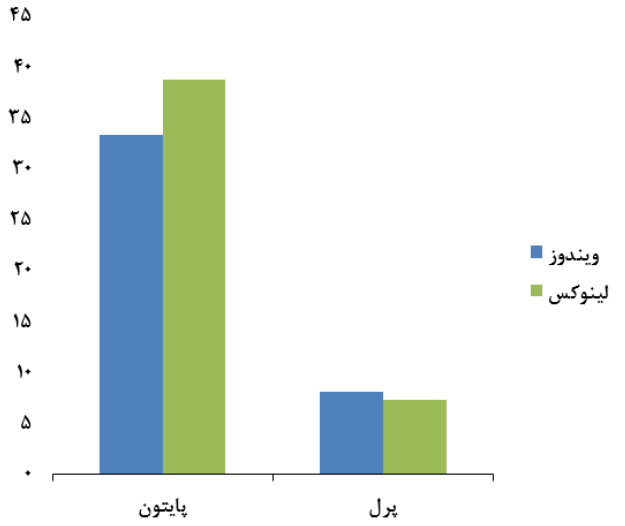
در این مطالعه به سادگی کدنویسی، به عنوان تعداد خطوط کدنویسی مورد نیاز برای نوشتن برنامه با توجه به در دسترس بودن کتابخانه‌ها، که یک معیار در تعیین تعداد خط مورد نیاز برای تفسیر کردن یک برنامه است مراجعه کرده‌ایم.

پرل به وضوح عملکرد بهتری نسبت به پایتون در عملیات ورودی/خروجی داشته است. هنگام خواندن یک فایل FASTA، پرل سه برابر سریع تر از پایتون، و به نصف فضای ذخیره‌سازی توالی‌ها در حافظه نسبت به پایتون نیاز داشته است (شکل ۲).

[۱] BioSQL
[۲] ASCII



شکل ۲: مقایسه حافظه مصرفی توسط الگوریتم‌های هم‌ترازی سراسری و اتصال همسایگی نوشته شده با دو زبان برنامه‌نویسی پرل و پایتون. برنامه در سیستم عامل‌های لینوکس و ویندوز اجرا شده است [۶].



شکل ۳: مقایسه سرعت برنامه تجزیه کننده BLAST نوشته شده با دو زبان برنامه‌نویسی پرل و پایتون. برنامه در سیستم عامل‌های لینوکس و ویندوز اجرا شده است. حجم فایل اجرا شده ۹۸ گیگابایت بوده است [۶].

با توجه به نتایج بدست آمده از برنامه‌های هم‌ترازی سراسری و اتصال همسایگی، پایتون نسبت به پرل از کارایی بالاتری برای اعمال تغییرات بر روی کاراکترهای رشته‌ای برخوردار است.

پایتون با صرف زمان بیش از ۳۸ دقیقه برای پردازش فایل در مقایسه با ۲۸/۷ دقیقه توسط پرل عملکرد ضعیف‌تری برای تجزیه یک فایل BLAST داشته است. این تفاوت تنها از ناتوانی پایتون در اداره فایل‌های بزرگ رخ نداده است، بلکه برای خواندن یک فایل بدون پردازش خطوط، پایتون ۳/۲ دقیقه زمان را در مقابل ۱/۴ دقیقه توسط پرل صرف کرد (شکل ۳).

پرل بر پشتیبانی وظایف نرم‌افزارگرا[1] به وسیله عبارات منظم موجود در برنامه، اسکن کردن فایل و ویژگی‌های تولید گزارش تاکید دارد در حالی که پایتون بر حمایت از روش‌های برنامه‌نویسی رایج مانند طراحی ساختمان داده‌ها و برنامه‌نویسی شیگرا تاکید دارد [۶].

۴ نتیجه‌گیری

نتایج این مطالعه نشان داد، پایتون یک زبان برنامه‌نویسی با کدنویسی ساده و پرل یک زبان اسکریپ‌نویسی قدرتمند است. هر زمان که نیاز به نوشتن یک اسکریپت سریع با استفاده از عبارات منظم باشد می‌توان از پرل استفاده کرد. پرل برای اتوماسیون کردن عملیات روی فایل‌ها و سیستم‌های فایل طراحی شده است. در حقیقت، در پرل برای استفاده از عبارت منظم یا فراخوانی دستورات سیستم، نیاز به انجام کار خاصی نیست، اما در پایتون برای انجام این عملیات به وارد کردن یک کتابخانه نیاز دارید. به دلیل اینکه بیش‌تر اسکریپت‌های پرل از توابع استفاده نمی‌کنند و زمانی که نیاز به استفاده از یک اسکریپت به بیش از یک‌بار داریم، بایستی از پایتون استفاده کنیم. هم‌چنین، به منظور استفاده از توابع و شی‌ها در اسکریپت، پایتون انتخاب مناسب‌تری نسبت به پرل است.

مراجع

[1] A. Bateman et al, The Pfam protein families database, Nucleic Acids Res., 32, D138–D141, 2004.

[2] D.A. Benson et al, GenBank. Nucleic Acids Res., 35, D21–D25, 2007.

[3] J. Chang, B. Chapman, I. Friedberg, T. Hamelryck, M. de Hoon, P. Cock, T. Antao, E. Talevich, B. Wilczynski, Biopython Tutorial and Cookbook, 2017.

[4] B. Chapman, J. Chang, Biopython: Python tools for computational biology, ACM SIGBIO Newsletter, 20 (2):15–19, 2000.

[5] J. Felsenstein, PHYLIP – phylogeny inference package (Version 3.2), Cladistics, 5, 164–166, 1989.

[1] Application-oriented

[6] M. Fourment, M.R. Gillings, A comparison of common programming languages used in bioinformatics, BMC Bioinformatics, 1471-2105-9-82, 2008.

[7] J. Tisdall, Beginning Perl for Bioinformatics, 384 pages, First Edition October 2001 ISBN: 0-596-00080-4, 2001.

[8] T. Kulikova et al, EMBL nucleotide sequence database, Nucleic Acids Res, 35, D16–D20, 2006.

[9] D.R. Maddison et al, an extensible file format for systematic information, Syst. Biol., 46, 590–621, 1997.

[10] W.R. Pearson, D.J. Lipman, Improved tools for biological sequence analysis, PNAS, 85, 2444–2448, 1988.

[11] J.E. Stajich, D. Block, K. Boulez, S. Brenner, S. Chervitz, C. Dagdigian, G. Fuellen, J. Gilbert, I. Korf, The BioPerl Toolkit: Perl Modules for the Life Sciences, Genome Research 12 (10):1611–gr.361602, PMC 187536, PMID 12368254, 2002.

[12] J.D. Thompson et al, CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice, Nucleic Acids Res., 22, 4673–4680, 1994.

[13] The UniProt Consortium, The universal protein resource (UniProt), Nucleic Acids Res., 35, D193–D197, 2007.

[14] E.M. Zdobnov, R. Apweiler, InterProScan – an integration platform for the signature-recognition methods in InterPro, Bioinformatics, 17:847-848, 2001.