Comparative Analysis of Ruby Language Libraries in the Field of Data Science

Elisabed Asabashvili
The University of Georgia
Tbilisi, Georgia
e-mail: z.asabashvili@ug.edu.ge

https://doi.org/10.62343/csit.2024.1

***Abstract* — As is known, there are over 700 programming languages used today. Each of them has its own advantages, disadvantages, and specific capabilities, so the choice of using one or another usually depends on its suitability for a particular task. The aim of this article is to discuss the capabilities of the Ruby language in the field of data science, as well as to present and analyze the advantages and disadvantages of using Ruby language libraries compared to other libraries.**

The Ruby language, its syntax, and features that have made it concise, predictable, and suitable for object-oriented programming have been influenced by Perl and Python languages.

A positive aspect of this language is the ability to accomplish the same task in multiple ways. Ruby programming language is cross-platform due to its interpretability; it can be run on any system and work equally well on all platforms. Thus, it is quite flexible and implements principles that are impossible in compiled languages.

***Keywords* — Ruby language, Data Science, Ruby language libraries.**

I. INTRODUCTION

Programming languages are ranked according to the TIOBE community index, which is an indicator of programming language popularity and is updated monthly. Popular websites such as Google, Amazon, Wikipedia, Bing, and more than 20 others are used to calculate the rating. It is important to note that the TIOBE index does not indicate the best programming language or the language in which most code is written, but it is mainly used to check the relevance of innovations in programming or make strategic decisions [1].

According to this ranking, Python was named the most popular language. The R programming language ranks 21st, Julia ranks 37th, and Ruby ranks 18th in the TIOBE index as of March 2024. It is worth noting that languages such as Julia, Scala, Ruby, MATLAB, Octave, SAS, and others, although not at the top, also deserve attention.

The Ruby programming language began to spread in Europe only in the 2000s. Initially, all Ruby documentation was in Japanese, which was a barrier for both Europeans and American scientists. Now, as we have already mentioned, it ranks 18th in popularity worldwide.

The Ruby language was created in 1995 by Japanese programmer Yukihiro Matsumoto (Matz), and its main philosophy was to create a language that is easy to use, convenient, and comfortable, while saving human time and effort.

Ruby's syntax and features that have made it concise, predictable, and suitable for object-oriented programming have been influenced by Perl and Python languages. Another positive aspect of this language is the ability to accomplish the same task in multiple ways, depending on what is convenient for the developer [2, 3].

Thanks to its interpretation, Ruby is a cross-platform language that can work on any system and performs equally well on all platforms. Therefore, it is quite flexible and implements principles that are impossible in compiled languages.

Although Ruby is primarily used as a server-side language, since it was created as a universal language, it can be used to write programs of any type.

Most often, the server-side part of websites and web applications is written in Ruby using the Ruby on Rails framework. Despite its popularity in this regard, this is not the only area of its use.

Some important programs are also written in Ruby, for example:

Metasploit - for penetration testing (pentest), which involves simulating the actions of a malicious actor trying to gain access to a user's information systems with the aim of compromising data integrity, confidentiality, or availability.

Vagrant - for working with virtual environments.

Homebrew - for installing applications on macOS via the command line.

In other words, code written in Ruby can be found in almost any development area [4].

II. USE OF PROGRAMMING LANGUAGES IN THE PROCESS OF WORKING WITH DATA

It is well-known that knowledge of data analysis/processing methods, probability theory, statistics, mathematics, as well as programming languages, forms the basis of data science. Using programming languages in data and database management processes is crucial because writing programs and scripts is necessary for data analysis, and there are many different programming languages available for this purpose [5].

These languages include languages such as:

Python - the most popular and versatile language. It is mainly used for web development, programming of intelligent devices, and API development. It is especially popular among people working with big data. Python is used to write artificial intelligence and machine learning programs, process large data using ready-made libraries and frameworks, etc. However, its drawback is that some operations are relatively

slower compared to some other languages, so it should be noted that this language may not be suitable if you need high speed. Additionally, it should be noted that in this program, a variable receives its type not during creation but during value assignment, which is undesirable and can lead to errors when working with data [5, 6].

R - is best suited for complex analytics, scientific research, and statistical data. It has tools for visualizing graphs, integration with databases, and support for machine learning methods. It contains thousands of ready-made functions but is difficult to master and practically unsuitable for solving other programming tasks. Moreover, knowledge of mathematical analysis, probability theory, and statistical methods is desirable to work with it. It works quite slowly when processing large data sets, which is due to its architectural features.

Java - is a versatile language used for working with big data, creating software, and applications. One of its main advantages is cross-platform compatibility. It is fast and very versatile compared to Python, but it is much harder to learn than other programming languages due to its complex syntax. It has fewer built-in tools than Python and R, and therefore is less useful for data analysis and processing. This language is not as widely used for data science tasks, but it should be noted that it has good libraries for working with data.

Scala - although not very popular and not even in the top 20 of the world ranking, it processes large data quickly thanks to parallel computing and is highly compatible with Java because this language runs on a virtual machine. The Scala programming language is not only difficult to learn but also challenging to read programs written in it, which is undoubtedly considered its drawback.

Go - created by Google specifically for analyzing and processing large data. It is mainly focused on extracting and analyzing information from databases, although it is also used for artificial intelligence and web development. The language is very simple and contains many standard libraries, but it is still very young and not fully matured, so it lacks some features and is not yet suitable for handling large projects. At this stage, programs are written for it only for individual microservices.

MATLAB - is a language designed for numerical computations, developed in the last century and intended for complex mathematical calculations and operations. It is more suitable for calculating indicators for data analysis rather than analyzing the data itself.

Julia - a new language specifically designed for working with data and closely resembling MATLAB. It is mainly used in machine learning and computer modeling. Like the Go language, it is not yet matured, so it has few ready-made functions and poorly structured libraries. It is worth noting that Julia is not an object-oriented language.

C++ - is a very fast general-purpose programming language, usually used for writing games, programs, etc., and is not intended for data analysis. As for data processing, tools such as MapReduce, the Caffe repository, or the Minerva neural network library are used for writing.

III. DISCUSSION OF RUBY LANGUAGE LIBRARIES AND THEIR CAPABILITIES

In a data-driven world, the importance of data science is enormous. Data is of utmost importance in today's digital world.

Thanks to the invention and advancement of mobile technologies – smartphones and tablets, mobile networks and Wi-Fi innovations – the creation and consumption of data is becoming more intense. As data grows, so does the need to process it.

The foundation of data science is knowledge of data processing methods, probability theory, statistics, coding and mathematics.

Querying data from a database, analyzing it, creating complex algorithms and running them using a neural network requires knowledge and use of programming.

Choosing effective programming languages and tools for solving Data Science problems is a very serious and responsible matter.

Data scientists should choose a programming language based on their own solution to the problems, choosing the most convenient of the listed languages.

The goal of this article is to discuss the capabilities of the Ruby language in data science and analyze the advantages and disadvantages of using this language compared to other libraries.

Unfortunately, the active use of this concise and elegant Ruby language is not observed in the field of data science, although it can be very successfully used for data analysis and machine learning.

For data science and machine learning, there is a set of libraries called SciRuby, and for statistical functions, there is DescriptiveStatistics. It is worth noting its integration with databases, specifically with SQL using Active Record and the Mongoid library for working with MongoDB [7].

Overall, the Ruby language is an excellent choice for scientific computations and data visualization. Although some languages may have more libraries, Ruby still has its place in science, and writing code in this language is always a pleasure.

It is worth mentioning that there are significant trends currently focusing on creating scientific computations in the Ruby language - the Ruby Science Foundation.

When it comes to data analysis, the focus is often on languages like Python and R, but there are also libraries in Ruby that make data analysis engaging and efficient. For example:

Pandas Library - This is the most popular Python library for convenient data manipulation and analysis. It uses a powerful tool called DataFrame (df) for this purpose. A DataFrame is a two-dimensional data structure where data is arranged in rows and columns. The Pandas DataFrame stores data in a tabular format, similar to Excel spreadsheets. Although this is a Python library, Ruby developers can access it with a shell after installation, allowing them to analyze data and write code in Ruby [8, 9].

Second frequently requested library NumPy Library - This library is mainly used for creating arrays or matrices and can be used with machine learning (ML) or deep learning (DL) models. Therefore, while Pandas is used for creating two-dimensional data objects, NumPy creates N-dimensional homogeneous objects.

It's worth noting that the Numo library in the Ruby programming language is similar to the NumPy library, and Daru serves as an equivalent to Pandas in the Ruby ecosystem.

Daru is a data analysis library written in Ruby with the tagline "Data Analysis in RUBy". Its data structure is a data frame, similar to an in-memory database table. Data frames consist of rows and columns, with each column assigned a specific data type. It's worth noting the role of the daru-view plugin, which is used in web applications and IRuby notebooks for data visualization, including simple and interactive charts. It can work in any Ruby web application environment, such as Rails, Sinatra, Nanoc, etc. [10]

Daru makes data manipulation simple and intuitive, primarily through two data structures: Daru::DataFrame and Daru::Vector. A vector is a basic one-dimensional structure similar to a labeled array, while a DataFrame is a two-dimensional structure resembling an Excel spreadsheet, used for managing and storing data collections.

Ruby users have access to several libraries, such as Pycall, which allows Ruby specialists to access Python functions. After installing Pandas, Pycall acts as a mediator between the Python and Ruby coding languages.

The ChartKick library is used for building charts in Ruby. Thanks to this library, it's possible to create histograms, pie charts, geographic diagrams, and more with just a single line of code.

The Nmatrix library is part of the SciRuby project, primarily written in C and C++, specializing in solving linear algebra problems. Nmatrix is often compared to the Panda library and the R language in terms of organizing data in matrices. It's worth mentioning that Ruby has the capability to perform simple regression, for which there is a well-known linear algebra library called SimpleLinearReprofit with several useful functions.

IV.CONCLUSION

Data science has found wide application in statistical and scientific computations, machine learning, data processing, storage, text and data analysis, as well as visualization.

As the discussion showed, some of them are more or less suitable for effective data processing due to the availability of a variety of ready-made tools, functions, and libraries. Therefore, to solve a specific data science task, it is necessary to choose a specific programming language, taking into account its pros and cons.

The goal of the article is to popularize the Ruby language in data science, discuss its capabilities, present and analyze the advantages and disadvantages of using this language compared to other libraries.

Unfortunately, the use of this concise and elegant language is not actively observed in the field of data science, although it can be very successfully used in data analysis, working with arrays, hashes, and machine learning.

REFERENCES

1. https://www.tiobe.com/tiobe-index/
2. J. Evans *Polished Ruby Programming*, Birmingham, UK, 2021
3. N. Rappin and D. Thomas, *Programming Ruby 3.3 (5th Edition)*, Publisher: Pragmatic Bookshelf, 2024
4. https://github.com
5. D. P. Kroese, Z. I. Botev, T. Taimre, R. Vaisman, *Data Science and Machine Learning. Mathematical and Statistical Methods*, CRC Press, 2023
6. E. Asabashvili. *Challenges of media digitization in Georgia*. XIII International Conference of the Union of Mathematicians of Georgia. Batumi, Georgia 2023.
7. SciRuby.com
8. E. Asabashvili. *Technological Challenges in Education.* 7th International Conference on Social Research and Behavioral Sciences. Antalya, Turkey, 2020
9. https://github.com/mrkn/pandas.rb
10. https://github.com/SciRuby/daru