Woody International Publish Limited
An Multidisciplinary Academic Journal Publisher

Journal of Artificial Intelligence and Information, Volume 2, 2025
https://www.woodyinternational.com/

Research on the Application of Python in Big Data Analysis

Tianyi Li

Wuchang Vocational College, Wuhan 430202, Hubei, China

Abstract: ***Python based development tools are of great significance for improving the analysis and processing of big data. Integrating the advantages of Python with its own strengths, optimizing and improving it from a "fast" perspective, and enhancing its efficiency in practice through visualization and data analysis. Under data programming, Python language can perform data processing in areas such as information acquisition and storage. Based on big data, it can enhance the ability of data processing and analysis.***

Keywords: Big data analysis; Python language; Application analysis.

1. Introduction

With the continuous development of information technology and the continuous improvement of science and technology, the 21st century will become a carrier of massive information. In today's "big data" social environment, the development of every discipline must rely on data analysis to achieve, rather than simply relying on "experience" or "intuition" to make decisions. Data processing is an indispensable part of big data research. Data processing is the use of appropriate statistical methods to organize the collected data, extract valuable information, and draw conclusions. The job of data analysis is to find valuable data from these large and chaotic data, give new meanings to these data, and provide some suggestions for decision makers.

In the field of 3D object recognition, Lyu et al. [1] proposed optimized convolutional neural networks (CNNs) for rapid 3D point cloud object recognition, demonstrating improved efficiency in processing complex spatial data. Similarly, Peng et al. [8] introduced a novel approach for 3D vision-language understanding using Gaussian splatting, which enhances the integration of visual and textual data in 3D environments. In the context of environmental technology, Liu et al. [3] explored the impact of supply chain digitization on the development of environmental technologies, highlighting the role of inflation and consumption in G7 nations. In healthcare, Li [4] utilized Bayesian optimization and deep learning to optimize clinical trial strategies for anti-HER2 drugs, showcasing the potential of AI in personalized medicine. Additionally, Tian et al. [14] improved brain tumor image segmentation by integrating the GSConv module and ECA attention mechanism into the Unet architecture, achieving higher accuracy in medical imaging. In the realm of finance, Bi and Lian [9] advanced portfolio management techniques using deep learning models, enhancing investment strategies through machine learning. Furthermore, Ren et al. [10] applied IoT-based 3D pose estimation and motion optimization for athletes, leveraging C3D and OpenPose for real-time performance analysis. In urban development, Zhou et al. [11] optimized an automated garbage recognition model using ResNet-50 and weakly supervised CNNs, contributing to sustainable urban management. In the domain of graph processing, Jin et al. [5] and Yang et al. [6] developed efficient FPGA-based subgraph matching and scalable subgraph enumeration systems, respectively, addressing the challenges of processing massive graphs. In network congestion control, Chen et al. [7] proposed Octopus, a system for in-network content adaptation to manage congestion on 5G links. Lastly, in the field of education and design, Xu et al. [13] introduced AI-enhanced tools for cross-cultural game design, facilitating online character conceptualization and collaborative sketching.

2. Python language and its features

Python is a widely used cross platform high-level programming language invented by Dutch mathematician Guido Van Rossum. The design philosophy of Python is that the code is easy to read and uses a simple and clear language. Since its official release in 1991, Python has undergone several modifications, added many new features, and has become increasingly simple and standardized. It has now been widely used in system management and web

programming. Python is a trump card for data analysis, and in 2017, it became the most popular programming language in a dazzling way, loved by programmers. Python has a large number of libraries and has shown great advantages in rapid development. In learning tasks such as data analysis, data science, and artificial intelligence, Python is the third largest language after C++ and Java.

The features of Python language are as follows:

1) Easy to learn Compared to other programming languages such as C, Python's programming is simple, easy to get started with, and suitable for beginners. For example, if we want to implement a specific function, Python requires only one tenth of the code required by C. However, writing in Python can greatly improve work efficiency, which is also the most popular aspect of Python.

2) In PythonObject, everything is an object. Both program orientation and object-oriented programming are supported. A program is built on an object that contains a large amount of data and functions. Compared to other mainstream languages such as C++ and Java, Python has extraordinary functionality and is also very easy to implement OOP.

3) Strong universality, as long as there are corresponding Python interpretation tools, Python software can work normally on various platforms such as Linux, Windows, Android, etc. The syntax of Python is concise and clear, making it very easy to use. Its grammar rules are clear, giving people a feeling of natural language and reducing the difficulty of learning. Python can run on multiple operating systems, including Windows, Linux, Mac, etc. Whether on a personal computer or a server, Python can be easily used for development.

4) The Python standard library is very powerful and can provide you with a large amount of information, such as regex expressions, databases, web browsers, XML, XML-RPC, HTML, WAV files, password systems, GUI, and other system related functions.

5) The Python standard code uses forced indentation to improve the readability of the code.

3. Analysis of the Advantages of Python Language

Firstly, Python syntax is concise and clear, and the code is easy to read and understand. Compared to other programming languages, Python has fewer lines of code, higher readability and maintainability, which helps improve development efficiency. Secondly, Python has a rich set of third-party libraries and modules, which can easily implement various functions such as data analysis, machine learning, network programming, etc. The existence of these libraries and modules greatly reduces the workload of developers and enables them to quickly implement complex functions. In addition, Python language can run on multiple operating systems, including Windows, Linux, Mac, etc. This enables developers to easily develop and run Python programs on different platforms, improving the portability of the programs. Finally, Python language can create a standard database, focusing on comprehensive control over databases, expressions, and other aspects. It also uses forced indentation methods to enhance code readability. In computer programming, commonly used languages include Java, C, Python, etc. There are also many types of languages, and Python is a relatively simple language that is easy to maintain and manage later programs. Its use is also quite common.

4. The Application of Python Language in Big Data Analysis

4.1 Establish Documents

It is a program that uses software tools to obtain webpage data. When designing a network crawler, information data can be obtained from information pages, filtered using LXML, and stored on a computer hard drive. Due to Python being an object-oriented programming tool, it is widely used for automatic design of mixtures. Due to the increasing diversity of programming techniques, Python's libraries have become more powerful. Python can be used independently or in combination with Djhango. Python itself has some unique features, and for specific use, we can use Python to implement if statements with indentation. The application of Python in practice can make data compilers more perfect and make data run more smoothly. Created a document, then performed corresponding operations on some important parameters in the document, and stored these parameters in the corresponding storage space.

4.2 Big Data Information Capture and Control

In the development of Python language and information processing, search engines can be used to provide an address. For example, after software development is completed, Baidu can be used to search for previous information and create a connection channel. Based on data analysis and data crawling, the rules and processing of data can be controlled to achieve data crawling. After setting up a new site, collaboration with other sites can be used to search using web crawling tools and add data extraction rules to complete data analysis and information processing.

4.3 Crawler Information Acquisition

Under the application of Python language, in order to obtain information on a page, web crawlers can be used to build search engines and analyze URL data, achieving the goals of data acquisition and information analysis. After obtaining this information, it can be compared and analyzed with the target information to find the corresponding URL information. After obtaining URL data, the data can be stored on a local hard drive and integrated with data information to enhance its analytical capabilities.

4.4 Ways of Storing Information

In big data analysis, Python language can store information in various ways, some of which commonly include: 1. Saving data in text form to a file, using Python's built-in function 'open()' to create, write, and read text files. Text files are the most basic storage method, suitable for storing data with simple structures. Excel files are a common format for storing and processing data, and Python can use the 'pandas' library to read, write, and process Excel files` Pandas provides flexible data structures (such as DataFrames) and powerful methods for processing and analyzing data. JSON is a lightweight data exchange format commonly used for storing and transmitting structured data. Python can use the 'json' module to process JSON files, parse JSON data into Python objects, or convert Python objects to JSON format for storage.

4.5 Data Preprocessing

When capturing web pages, they contain a lot of advertisements and photos, and it is likely to cause data distortion during the capture process. In the Python language, there are various libraries and tools available for data preprocessing. Here are some commonly used data preprocessing techniques and corresponding Python libraries: 1. Data cleaning: Data cleaning refers to handling errors, missing values, outliers, and other issues in data. The 'pandas' library in Python provides powerful data cleaning features, such as removing duplicate values, filling in missing values, replacing outliers, and more. 2. Data conversion: Data conversion is the process of converting data from one form to another, such as data type conversion, data encoding conversion, data format conversion, etc. The pandas and numpy libraries in Python provide rich data conversion functions and methods, making it easy to handle data conversion needs. By utilizing big data analysis and processing, it is possible to achieve the level of data analysis and processing in Python language.

4.6 Page Search

Firstly, it is necessary to connect to the database. Secondly, write query statements to query data based on the required conditions and sorting rules. By executing the query statements, all data that meets the conditions can be obtained. Afterwards, the amount of data displayed on each page and the current page number can be determined based on the needs of pagination. When the amount of data is very large, pagination retrieval may cause performance issues. To improve query efficiency, methods such as database indexing and optimizing query statements can be considered.

5. Design of Big Data Analysis in Python Language

According to the characteristics of Python, data information processing can be carried out, which improves the comprehensive level of data reading [2]. On the basis of data logic analysis and information processing, Python language can be used to optimize file information and editing processes, and to enhance the practical value of Python language in data analysis by managing and controlling file information. In the documentation created in Python language, instruction logic can be used to process the data in the document. After editing the information in the document, numerical operations and the seal interface can be optimized. Firstly, it is necessary to determine the source of big data to be analyzed. It can be databases, log files, API interfaces, etc. Use third-party libraries in

Python, such as Requests and BeautifulSoup, to retrieve data and store it in appropriate data structures, such as lists or data boxes. Secondly, clean and preprocess the obtained data for subsequent analysis. This includes handling missing values, handling outliers, data type conversion, etc. The Pandas library in Python provides convenient data cleaning and preprocessing capabilities.

Therefore, by making reasonable use of these tools and libraries, big data analysis can be efficiently and accurately conducted to discover valuable insights and solutions. And it can comprehensively control parameters and assignment changes, thereby completing the display and analysis of data. By using cmd instructions, data can be called and processed, achieving data mining and information processing.

6. Basic Tools of Python in Big Data Analysis

6.1 Introduction to Python Programming Language

6.1.1 The features and advantages of Python

1) Python is simple and clear: Python's syntax construction is clear and easy to understand, reducing program complexity. 2) Massive databases and models: Python includes databases and models for massive data processing, statistical analysis, and machine learning, which can provide strong technical support for big data analysis and analysis. 3) Cross platform: Python is well compatible with various applications, making it easy to perform data analysis on various applications. 4) Effective Development: Python has a fast development speed and debugging speed, which can help data experts and analysts quickly establish and test it.

6.2 Python Libraries and Tools

6.2.1 NumPy and Pandas: Data Processor Mass

NumPy is a very important library in Python, which provides effective means for multidimensional array operations and numerical calculations. It can also perform broadcasting, vectorization, and other operations, accelerating the processing and calculation speed of massive data. 1.2 Pandas is a data analysis library developed based on NumPy, which provides a high-performance, flexible, and simple data structure, as well as some data processing methods such as Series and DataFrame, providing convenience for data cleaning, organization, and transformation.

7. Conclusion

In terms of big data analysis, using Python programming language, through the research of this project, it is possible to effectively mine big data and improve the ability of big data analysis and information processing. When using Python, reading output files in XML format can enhance the practical application of Python in big data analysis through the design and use of web crawlers. In information processing, it includes files, execution, scripts, and real-time information. By utilizing these contents to control the data output process and information processing, it achieves the management and control of output and corresponding data to meet the practical needs of big data analysis and processing.

References

[1] Lyu, T., Gu, D., Chen, P., Jiang, Y., Zhang, Z., Pang, H., ... & Dong, Y. (2024). Optimized CNNs for Rapid 3D Point Cloud Object Recognition. arXiv preprint arXiv:2412.02855.

[2] Yin, Y., Xu, G., Xie, Y., Luo, Y., Wei, Z., & Li, Z. (2024). Utilizing Deep Learning for Crystal System Classification in Lithium - Ion Batteries. Journal of Theory and Practice of Engineering Science, 4(03), 199–206. https://doi.org/10.53469/jtpes.2024.04(03).19

[3] Liu, H., Li, N., Zhao, S., Xue, P., Zhu, C., & He, Y. (2024). The impact of supply chain and digitization on the development of environmental technologies: Unveiling the role of inflation and consumption in G7 nations. Energy Economics, 108165.

[4] Li, T. (2025). Optimization of Clinical Trial Strategies for Anti-HER2 Drugs Based on Bayesian Optimization and Deep Learning.

[5] Jin, X., Yang, Z., Lin, X., Yang, S., Qin, L., & Peng, Y. (2021, April). Fast: Fpga-based subgraph matching on massive graphs. In 2021 IEEE 37th international conference on data engineering (ICDE) (pp. 1452-1463). IEEE.
[6] Yang, Z., Lai, L., Lin, X., Hao, K., & Zhang, W. (2021, June). Huge: An efficient and scalable subgraph enumeration system. In Proceedings of the 2021 international conference on management of data (pp. 2049-2062).
[7] Chen, Y., Tahir, A., Yan, F. Y., & Mittal, R. (2023, December). Octopus: In-Network Content Adaptation to Control Congestion on 5G Links. In 2023 IEEE/ACM Symposium on Edge Computing (SEC) (pp. 199-214). IEEE.
[8] Peng, Q., Planche, B., Gao, Z., Zheng, M., Choudhuri, A., Chen, T., ... & Wu, Z. (2024). 3d vision-language gaussian splatting. arXiv preprint arXiv:2410.07577.
[9] Bi, S., & Lian, Y. (2024). Advanced portfolio management in finance using deep learning and artificial intelligence techniques: Enhancing investment strategies through machine learning models. Journal of Artificial Intelligence Research, 4(1), 233-298.
[10] Ren, F., Ren, C., & Lyu, T. (2025). Iot-based 3d pose estimation and motion optimization for athletes: Application of c3d and openpose. Alexandria Engineering Journal, 115, 210-221.
[11] Zhou, Y., Wang, Z., Zheng, S., Zhou, L., Dai, L., Luo, H., ... & Sui, M. (2024). Optimization of automated garbage recognition model based on resnet-50 and weakly supervised cnn for sustainable urban development. Alexandria Engineering Journal, 108, 415-427.
[12] Fan, Y., Wang, Y., Liu, L., Tang, X., Sun, N., & Yu, Z. (2025). Research on the Online Update Method for Retrieval-Augmented Generation (RAG) Model with Incremental Learning. arXiv preprint arXiv:2501.07063.
[13] Xu, Y., Shan, X., Lin, Y. S., & Wang, J. (2025). AI-Enhanced Tools for Cross-Cultural Game Design: Supporting Online Character Conceptualization and Collaborative Sketching. In International Conference on Human-Computer Interaction (pp. 429-446). Springer, Cham.
[14] Tian, Q., Wang, Z., & Cui, X. (2024). Improved Unet brain tumor image segmentation based on GSConv module and ECA attention mechanism. arXiv preprint arXiv:2409.13626.

Author Profile

Tianyi Li (1989.10-), female, Han, Xiangyang, Hubei, master's student, lecturer, research direction: information technology and curriculum integration, computer applications.