

# Design and Implementation of an Online Python Teaching Case Library for the Training of Application-Oriented Talents

<https://doi.org/10.3991/inet.v15i21.18191>

Shunye Wang, Dayong Liu, Ning Wang  
Langfang Normal University, Langfang, China

Yanxiang Yuan (✉)  
Qiannan Normal University for Nationalities, Duyun, China  
yanxiangyuan12@163.com

**Abstract**—Based on the features of Python and the situation of case-based teaching, this paper develops and implements a complete online Python teaching case library. The teaching contents of a Python programming were divided into ten parts: basic knowledge, three basic structures, functions and modules, object-oriented thinking, file and database programming, network and multithreading programming, image programming, web crawling, as well as data analysis and visualization. Each part was provided with several basic teaching cases. Following the idea of software engineering, our case library was designed based on ChaoXing, a famous online learning platform in China. The designed case library offers a clear path from data acquisition, data storage, data processing, to data visualization. It also provides a wealth of up-to-date and extensible contents, enabling learners to apply the knowledge to practice. Through application of the proposed case library, the students all achieved excellent performance, published innovative academic papers, and selected suitable topics for graduation thesis. The library also greatly promotes the abilities of students in computational thinking, programming level, and solving complex engineering problems.

**Keywords**—Python, case-based teaching, case library, teaching practice

## 1 Introduction

Python is an interpreted and object-oriented scripting language with dynamic semantics and beautiful syntax. Due to the simplicity, flexibility, and ecological completeness, it has been widely used [1-3]. At present, 98% of artificial intelligence is developed using the Python programming [4], which focuses on problem-solving methods, and the analysis for the logic and algorithms of the program itself [5]. Many colleges and universities at home and abroad offer relevant courses not only in the majors closely related to information technology such as computer, software engineering, and big data etc., but also in many others involving scientific computing such

as mechanics, biology, management, mathematics, and accounting, etc. [6]. Even some open Python courses to support the frontier teaching of the subject in the fields of social sciences and journalism.

Case-based teaching is one of the effective methods for teaching language [7-9]. In order to stimulate students' interest, Huang and Song [7] presented the Python case in a graphical way, and proposed a graphical case-based teaching plan. Zhao [8] put forward a program design teaching mode integrating general education and professional education for the inter-disciplines, and introduced the case teaching methods related to the material major based on the teaching practice of this major. Song and Huang [9], based on the full computing ecosystem of Python, gave a few interesting and useful teaching cases, greatly enriching the system and content of case teaching. Compared with other completed case libraries of program language constructed for a long time [10-14], the online case-based Python teaching methods and effects have been mostly studied [15-17], but there are relatively few studies on the construction of related case library and a lack of case resources.

Introducing the idea of software engineering into the Python programming course, this paper decomposes the solution of large-scale project cases into several basic cases, explains the realization process separately, and then combines them to construct a problem-driven, step-by-step teaching process, thereby improving students' analysis of problems and the ability to solve problems using Python programming. In the teaching process, a mind map [18] was used to explain technical principles, the context of knowledge, and the process of realization.

## **2 The Content of Python Courses**

Through careful study of various teaching materials related to Python programming [19-22], it is determined that the teaching content of the Python programming language mainly consists of 10 parts as follows:

1. Basic knowledge of Python, mainly including the basic grammar, and basic data types such as lists, tuples, dictionaries, sets, and string operations.
2. Three basic structure. The sequence is relatively simple; the focus is on the application of selection and loop.
3. Functions and modules. It mainly includes the design and use of functions in practice, how to realize complex functions and code reuse through modular programming, and the installation, introduction, and application of common modules.
4. Object-oriented thinking, mainly referring to the main ideas of oriented object and class, through practice to understand class encapsulation, inheritance, and polymorphism, and realize class reuse.
5. File programming, i.e., the access and operation of files and folders, especially the reading and writing of txt, csv, excel files, to prepare for data access.
6. Database programming, i.e., the realization of the addition, deletion, modification, and checking functions of SQLITE and MySQL databases, and the application as a background data storage function.

7. Network programming and multithreading, mainly including TCP and UDP programming, the realization of email service applications, and the multi-thread programming in the operating system.
8. Image programming, i.e., the application of Turtle, Graphic and Tkinter packages to realize the design of graphical interface and the generation of interesting images.
9. Web crawling. It mainly refers to the principle of web crawler, through the introduction of Urllib2, Requests, BeautifulSoup, Scrapy packages to achieve network data crawling, especially the application of regular expressions.
10. Data analysis and visualization, that is, the application of Numpy, Pandas and Matplotlib packages to realize data analysis and visual representation, and improve the interpretability of analysis results.

### **3 Python Teaching Case Library Design**

For traditional programming courses, the experimental teaching is synchronized with the content of the textbooks, which limits the experimental content and students' thinking. Especially when the students have the foundation of C language programming and basic programming thinking, stereotyped content structure in ordinary textbooks should be discarded when learning the Python language. It should spend less time teaching and practicing basic grammar and basic structure, but focuses on the later application development.

The case library was designed based on ChaoXing, a famous online learning platform in China. The courses were conducted according to the ten parts of the content in the syllabus. Each part is provided with basic teaching cases, and those with high designability and comprehensiveness. The case library should be designed based on the following principles:

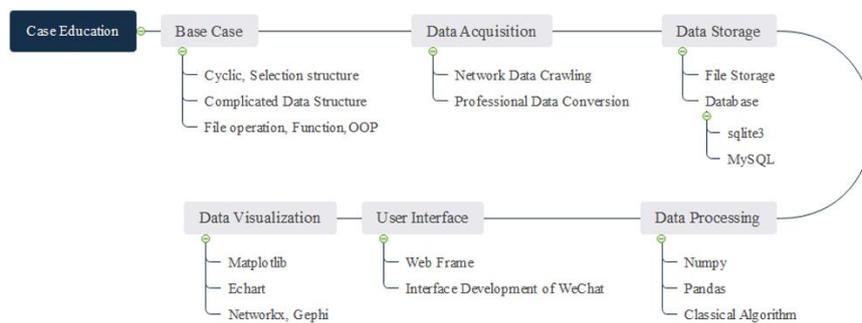
#### **3.1 Gradual improvement and clear path**

Considering that the students have certain programming thinking and experiences, the teaching content is developed in accordance with their professional cognition, and the thinking of teaching practice should be changed, i.e., clarifying the research ideas of various problems for students from the perspective of software engineering. Thus, for the engineering practice problems raised, the students can perform scientific analysis independently or through cooperation by group, actively learn relevant knowledge and technology to explore solutions to problems, and achieve the optimization. On this basis, the case library specifically includes 6 modules: basic case, data acquisition, data storage, data processing, data visualization, and user interaction, as shown in Figure 1, which offers a path for comprehensive teaching cases.

The content of teaching practice includes not only basic cases for realizing single knowledge for each module, but also comprehensive application cases with strong comprehensiveness and designability. They include the calculation of PI value of the training loop structure and the calculation of the butterfly effect reflecting the compound interest effect, and so on. The function module includes the classic program of

the Tower of Hanoi and the man-machine tic-tac-toe game, and so on. The student management system is still retained in the information system training. In the crawler practice, there are basic cases such as Chinese university ranking crawlers, Baidu image crawlers, social platform network comment crawlers, and campus network search engines. The base case is the cornerstone of the comprehensive case.

The comprehensive case was developed in the execution process of engineering practice. It's gradually implemented and improved following the process of data acquisition, data storage, data processing, algorithms and applications, and data visualization. Thus, a comprehensive project case with higher requirements is finally completed based on the development experience of basic case in each module.



**Fig. 1.** Python case-based teaching path

In the comprehensive practical case of "sentiment analysis of a certain product user satisfaction on an e-commerce platform", it's necessary to first crawl all the comment information of users who have purchased the product on the e-commerce platform through crawler technology, and then preprocess the comment data, that is, the text deduplication, text cleaning, word segmentation, and removal of stop words. Afterwards, sentiment orientation analysis was performed on the pre-processed data. Finally, the model was continuously optimized to achieve the best results, which were displayed and fed back. Different word segmentation algorithms can be used, such as string-matching algorithms, hidden Markov models, etc. Besides, dictionary-based sentiment analysis methods were used and LDA topic models were constructed, to finally obtain users' positive and negative evaluation of the product, and then feedback to the producers and sellers to improve the quality and services.

### 3.2 Applying the acquired knowledge to practice and keeping up with the times

The goal of application-oriented talent training is to apply. The training concept of new engineering education is to be student-centered and output-oriented for continuous improvement. Only by student development as the prerequisite, and applying the knowledge acquired to engineering practice, the students can make progress, keep pace with the times and seek innovation.



times can enrich students' cognition, stimulates their enthusiasm for learning, applies the knowledge in daily life, and cultivates a sense of social service.

### 3.3 Rich content and strong extensibility

The teaching case library should be constantly innovated and updated in the teaching process (Figure 3). The innovation cases of teachers and students should be increased, which not only reflects the deep connotative construction of computer science courses, but also highlights the extensional construction of computer technology in other disciplines to enhance students' sense of professional identity and social service awareness.

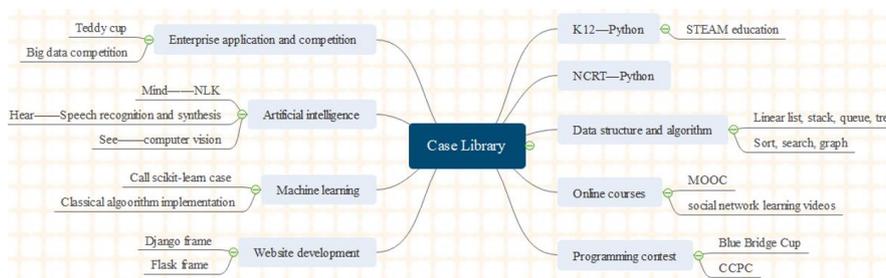


Fig. 3. Online Python teaching case library

Case library cover the full computing ecosystem of Python programming, mainly including K12 programming, computer rank test NCRT (Python language), data structure and algorithm, other online courses, programming contest, Website development, machine learning, artificial intelligence, and enterprise applications and competition.

Since the major of computer science and technology in our university is a teacher-training major, students also have the responsibility to teach in K12. In order to cultivate the ability of computational thinking of the primary and middle school students, programming education courses in primary and secondary schools have been gradually opened. And Python courses are set in higher grades. In this context, we added interesting and useful cases, set up graphic and game application cases to cultivate students' STEAM academic ability and innovation. The case-based teaching and game teaching methods can improve the computational thinking ability of young people [23-24].

In the process of knowledge learning, examinations and competitions are effective means to promote and test the effect of learning. The case library covers the contests of Python from easy to difficult, namely the NCRT, the Programming Competition, and the competition for the actual problems of the enterprise. Due to the low requirements, the computer rank exam NCRT has a high pass rate, which mainly examines the basic grammatical rules of the Python language and reading programs. So, it's enough to understand the third-party packages in the Python computing ecosystem and master at least two. The case of the programming competition mainly involves the

exam questions and simulation questions in Hebei Trial of the Blue Bridge Cup Programming Competition and the National College Student Programming Competition. This type of competition puts forward higher requirements on students' computational thinking and programming ability, and focuses on the examination of data structures and algorithm part. Such case is representative. The competition cases involving the enterprise engineering practice mainly refer to the cases of past exam paper in the Teddy Cup Data Mining Challenge and other mathematical modeling and big data competitions. Such cases are designed to solve practical problems in production and living. They not only need to consider the realization of functions at higher accuracy and efficiency, but also need to write documents. It is important to train the ability of solving complex engineer.

Finally, case library also includes certain practical projects in Python-based follow-up courses such as Web site design, machine learning, and artificial intelligence. These cases are mainly for students who have spare time to practice and innovate. They can call related packages to implement [25, 26], or implement complete algorithms in their own code. These courses are scheduled for follow semesters. If one wants to participate in the competition in the freshman year and sophomore year, they need to understand the related knowledge in advance in the case library, and prepare for the competition with teacher guidance and self-study. Although these cases are somewhat difficult, they are indeed an indispensable part of building a complete picture of the Python computing ecosystem.

#### 4 Case-Based Teaching Practice

Case-based teaching was conducted in the Python programming courses for the 2015 software engineering majors and the 2016 and 2017 computer science and technology majors. Students learned the basics of Python language through the flipped classroom, and extended from the basic cases to comprehensive cases. In the class, they were issuing the questions, analyzing and then solving problems step by step. Then must pay attention to problem expression and focusing on document writing.

In case-based teaching, students' computational thinking ability and innovation consciousness are also cultivated. A group of practice cases of outstanding student have emerged. They presided over and participated in some innovative projects, and published related papers based on the results of practical training (Table 1).

**Table 1.** Innovation cases of outstanding students

lost value	Linux code	Dialogue
lost value(n=100)	3.9481 to 3.1302	3.4519 to 2.6396
lost value(n=1000)	3.9548 to 2.1446	3.4513 to 2.2066
lost value(n=1000)	3.9306 to 1.9194	3.4384 to 1.9771
lost difference value(n=100)	0.8179	0.8123
lost difference value(n=1000)	1.8102	2.0112
lost difference value(n=10000)	2.0112	2.0112
Average Lost	1.54643	1.1529

Li et al. [27] discussed the characteristics of unstructured databases, the MongoDB database and its advantages of retrieval, and demonstrated the connection of MongoDB. The experiment proved that MongoDB's MongoClient connection is more convenient and faster, the Python-based connection is easier than Java connection. When using Gridfs storage, large files are easy to display. It is recommended to use when storing large files.

Zhao et al. [28] researched, analyzed and sorted out the comments of Baidu Post bar which is from 20 universities in the Beijing-Tianjin-Hebei Region, then applied the Single-pass algorithm, used the IK Analyzer word segmentation toolkit to calculate the text similarity and perform the repeated matching to complete the text mining of all comments. This shall help to find students' concerns, which is convenient for universities to strengthen public opinion analysis and management, and improves school management, and strengthens system construction.

Feng Shilong et al. described a public opinion analysis method based on self-media video platforms in an accepted paper, crawled the comment data in the current mainstream self-media video platforms such as Bilibili and TikTok using web crawlers, and performed sentiment and significant analysis for the keywords after Chinese word segmentation. This can reflect the public opinion of the self-media video platform, propose the meaning and methods of public opinion analysis, and can better assist the government cultural department to monitor public opinion on the Internet.

Zhao Zihua et al. designed a smart dressing APP in an accepted paper. First, she builds a database of users and clothing, and used a user-based collaborative filtering recommendation algorithm to mine and recommend similar users' wearing preferences. Then, she crawled the weather data to perform a new recommendation. The last recommendation effect is more accurate when combining with the weather conditions of the day. Finally, the clothes that are not in the database were linked to the shopping platform and completed a high-quality private exclusive service, which fully satisfied the individual needs of different users for smart dress.

Also, in the selection of the graduation project for the senior year students, Python language also is regarded as their first choice. They can conduct in-depth research on cases in the course teaching.

A word analysis system based on the original corpus made by a student applied the Word2Vec model to Chinese and English respectively through the deep learning TensorFlow framework, converted the words into vectors, and input them into the recurrent neural network. In this way, the neural network was trained and finally generated the more readable text. The students collected a piece of Linux code and a piece of English dialogue from the Internet and set the training times of 100, 1,000, and 10,000 rounds respectively for verification. The training effect is shown in Figure 4 and 5. Based on the running log of each training result, the change of experimental loss value was found, as shown in Table 2. Experiments show that as the number of training increases, the loss value gradually decreases, that is, the model training result is closer to the true value; at the same time, as the amount of training text increases, the training effect is better.

```

* t nns_ssnt nrins_stesitraser tint *tint tat st sns_cennre_ceses nt sract ste tr set nt
nt_sect crare cacn ctrate cntate ** n *

*t tens st ractacasnrer cost stacast te s nt tos snt te *hast re ns tt rastarocn
ttre cont cerocate tote_sticetinne *tasre *itre_rarotant_setet stint te tet

```

(a) Training Num=100

```

bieds in size, cont it the fite to the the tring conse. */
if (cles_reac_cont)
{
    cpu(&trace_revent_task_trine(size(trace_irq, &schode, to struct struct rq-
>curr_task)
{
    iff(tigt *rq->this_it_trock_irq_toime_file);
    return 0;
    if (id_file_spand_indatic_seq_trace_cfs_troce_clock_seq);
}
}
chat =

```

(b) Training Num=1000

```

-> seq, trace_pass, &trace_cpus, ftrace_perf_stack);
if (!pid_proc(struct filter_task_struct func = &cpu_spuc) {

if (count_sa_set,
    .page = start_procs0 & RB_PIRTALL_SESTIT_SYMONE;
    if (!sigeat);
    return 0;
    if (!traceper_page_dest))
}
static inlon iter = freq_perf_dome(&dlaper, &set, stat_t

```

(c) Training Num=10000

Fig. 4. Linux Code

```

<unk>sher te mos to toris the ho the se and are me mind sor sash shat
Ther ther to mashe me mise me mal thart an tind ane sero ath terot oor, were hate
hins, er aten tit sos soris orient
tot ondit at alls serestes ar thas ardesd, tither tas arer to herases sotens found an shor tin
det
I hald tato hand the

```

(a) Training Num=100

```

pand as that wing of shich, a my my love aroth and sellen to thee shill the stold, I'll hands;
and home one to hasted sir,
This weres so whing weld trust will she trown
Wite are stold to weald,
In thit, and my torned treen a dise of well worl see tome oft of and sonest
To wat of hath and sene as ston

```

(b) Training Num=1000

```

pent.
PESTER:
In sake hath have say, the was as a mound thy weld so me and he see his worsser.

CEETIO:
Therefring son out; as sir; who more.

MERETANA:
That, what to his thought on they wished or me
To me, than which some well, to my lord to my wits is their will.

LALOSANDUR:
I death the fair a
    
```

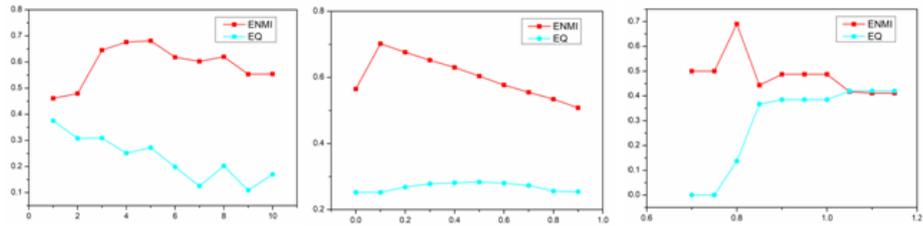
(c) Training Num=10000

Fig. 5. Dialogue

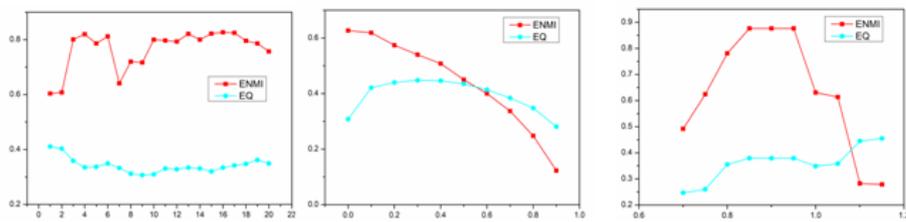
Table 2. Comparison of loss values in neural network case training

Author (Student)	Article title	Journal	Funds
Jiwei Li etc.	The Storage Technology in NoSQL of MongoDB	Computer Knowledge and Technology	Innovation and entrepreneurship training program for college students in Hebei Province (2017)
Fen Zhao etc.	Analysis on the Network Public Opinion of University students	Computer Knowledge and Technology	Innovation and entrepreneurship training program for college students in Hebei Province (2017)
Shilong Feng etc.	Analysis about the Network Public Opinion of Self-media Platform	Modern Computer (Employed)	Innovation and entrepreneurship training program for college students in Langfang Normal University (2019)
Zihua Zhao etc.	Design of Collaborative Filtering Recommendation of the Intelligent Dressing APP based on Meteorological Data	Journal of Langfang Normal University (Natural Science Edition) (Employed)	Innovation and entrepreneurship training program for college students in Langfang Normal University (2019)

The topic of another student’s graduation thesis was community discovery. It found the CORPA and SLPA algorithms through the overlapping community discovery LPA algorithm on the Karate and Dolphin data sets, and another LFM algorithm based on local expansion. EQ and ENMI indicators were analyzed to evaluate the pros and cons of community discovery algorithms. The case results are shown in Figure 6 and Figure 7. The study found that the SLPA algorithm has higher requirements for the selection of parameters and it is not easy to choose the termination conditions; the COPRA algorithm is not as sensitive to parameters as SLPA, and the accuracy of this algorithm is better than the SLPA algorithm in small-scale networks.



**Fig. 6.** EQ and ENMI parameter performance analysis of COPRA, SLPA, LFM algorithm in Karate data set



**Fig. 7.** EQ and ENMI parameter performance analysis of COPRA, SLPA, and LFM algorithms in the DOLPHIN data set

There are also some students who selected topics with highly engineering value such as the sentiment analysis of the barrage in self-media platform and complexity analysis of public transportation network, which shows a good effect. Through continuous tracking of student development and interviews, students all expressed their preference for case-based teaching methods. Especially for complex engineering project cases, their programming level has been improved, no longer afraid of complex problems, and the function analysis of graduation design is more comprehensive, accurate, smooth, and objective-oriented.

## 5 Conclusion

Case-based teaching is a commonly method used in programming language teaching. Students can understand algorithm thought and solve practical problems through cases education. For students with programming foundation, they apply the flipped classroom teaching mode to learn Python language, learn basic knowledge after class, analyze basic cases and comprehensive cases in class, explore knowledge points, decompose functions, and gradually implement and test the code, which can increase student participation, and enhance their interest.

The existing teaching case library is designed in accordance with the principle of gradual improvement and clear path, applying the acquired knowledge to practice and keeping pace with the times, and rich content and high extensibility. Due to the limited of class hours, students can learn cutting-edge cases they are interested in by themselves, and prepare for various competition, innovative projects, and graduation

design in advance. Through the feedback of teaching practice, it can be found that students are satisfied with the cases, get more motivated to learning, and have a more in-depth understanding of knowledge application and higher programming ability. They achieved good experience in projects, competitions, and graduation theses.

Next, the design of case library should be still students-centered, and enriches the resource library, especially by increasing the engineering practice cases with high designability and comprehensiveness, which is specifically reflected in the following aspects:

1. Focus on extensively developing personalized teaching and interdisciplinary training, then share professional resources with various disciplines such as sociology, bioinformatics, e-commerce, intelligent transportation, network communication, intelligent education, literature, and art, etc. The aim is to strengthen the extensional development of computer technology in various application fields.
2. Strengthen the echelon construction of the team of teachers, and enhance the teaching research by combining the scientific research results of the teachers with teaching cases to improve the level of the cases, and increase the difficulty of challenges.
3. Explore the cases of Python language-based programming education and artificial intelligence education in primary and secondary schools, and increase the duties and capabilities of the social service, which shall provide resources for information technology teaching of these schools.

## **6 Acknowledgement**

This paper is supported by a number of teaching and research projects: Exploration and Practice of Optimizing Software Engineering Curriculum System for Application-oriented Talent Training in Cooperative education project of industry university cooperation of Higher Education Department of Ministry of Education (No.201701017021); Programming Case Library Platform Construction for the Training of Applied Innovative Talents in the Research and Practice Project of Higher Education Reform in Hebei Province in 2018-2019 (No.2018GJJG348); Natural Science Foundation of Guizhou Province (No. KY[2016]316); Qiannan Science and Technology Planning Project (No. (2018)10); Project-driven Programming Language Reform in the Project of Education and Teaching Reform in Langfang Normal University (No.K2017-17)); First class curriculum construction of "Python Programming" in Langfang Normal University; Qiannan Normal University for Nationalities Foundation (No. qnsy2018017).

## 7 References

- [1] Luo, X., Ren, Y., Shan, X.M. (2004). Python based mixed-language programming and its implementation. *Computer Applications and Software*, (12): 17-18, 112.
- [2] Ying, F.Q., Zhang, Z.Y. (2019). Data visualization analysis of big data recruitment positions in Hangzhou based on python. *Review of Computer Engineering Studies*, 6 (4), 81-86. <https://doi.org/10.18280/rces.060403>
- [3] Neelapu, R., Devi, G.L., Rao, K.S. (2018). Deep learning based conventional neural network architecture for medical image classification. *Traitement du Signal*, 35(2), 169-182. <https://doi.org/10.3166/ts.35.169-182>
- [4] Han, Z.H. (2018). Exploration of Python teaching in the context of artificial intelligence. *computer era*, (7): 76-81.
- [5] Liu, W.G., Kui, X.Y. (2017). Construction of Python programming teaching system. *Computer Education*, (8): 13-17.
- [6] Schoute, M. (2019). Teaching Python to management accounting students: an illustration using support department cost-allocation methods. *The Accounting Educators' Journal*, 29(1): 137-161.
- [7] Huang, T.Y., Song, T. (2017). Python case teaching method and practice of drawing interest with graphics. *Computer Education*, (8): 32-37.
- [8] Zhao, G.H. (2017). Case teaching of cross-fusion Python programming for new engineering. *Computer Education*, (8): 23-27.
- [9] Song, T., Huang, T.Y. (2017). New thinking on teaching cases of Python Programming Language. *Computer Education*, (12): 11-14, 19.
- [10] Bais, H., Machkour, M. (2019). Method and apparatus for querying relational and XML database using French language. *Revue d'Intelligence Artificielle*, 33(6), 393-401. <https://doi.org/10.18280/ria.330601>
- [11] Saddhono, K., Hasanudin, C., Fitrianiingsih, A. (2019). The ability to think creatively on SSCS using schoology Apps, how is the student's language metacognitive awareness? *Ingénierie des Systèmes d'Information*, 24(4), 367-375. <https://doi.org/10.18280/isi.240402>
- [12] Liu, W.T., Jia, Y., Dong, Z.Y., Zhang, L. (2017). Research on practical teaching of software engineering specialty based on java case library. *Journal of Wuhan Polytechnic University*, 36(2): 110-113.
- [13] Masoumi, S., Mahjur, A. (2019). Collaborative component interaction. *Ingénierie des Systèmes d'Information*, 24(3), 321-329. <https://doi.org/10.18280/isi.240312>
- [14] Prastikawati, E.F. (2019). Dyned programme as computer assisted language learning (CALL) for university students: A perception and its impact, *International Journal of Emerging Technologies in Learning*, 14(13), 4-20. <https://doi.org/10.3991/ijet.v14i13.10448>
- [15] Klimeková, E., Tomcsányiová, M. (2018). Case study on the process of teachers transitioning to teaching programming in Python. *International Conference on Informatics in Schools: Situation, Evolution, and Perspectives*, 216-227. [https://doi.org/10.1007/978-3-030-02750-6\\_17](https://doi.org/10.1007/978-3-030-02750-6_17).
- [16] Faqih, B., Daoudi, N., Ajhoun, R. (2018). Design of an intelligent educational resource production system. *International Journal of Emerging Technologies in Learning*, 13(12): 4-18. <https://doi.org/10.3991/ijet.v13i12.8914>
- [17] Lu, B.B., Meng, X.L., Yuan, T.T., Wang, L.J., Zhu, S.S., Zhang, C.J. (2020). Case-based experimental teaching of Python language programming course. *Computer Education*. (4): 80-83
- [18] Astriani, D., Susilo, H., Suwono, H., Lukiati, B., Purnomo, A. (2020). Mind Mapping in Learning Models: A Tool to Improve Student Metacognitive Skills. *International Journal*

- of Emerging Technologies in Learning, 15(6): 4-17. <https://doi.org/10.3991/ijet.v15i06.12657>
- [19] Liu, Y.Z. (2017). Python3.5 learned from Zero, Tsinghua University Press.
- [20] Xia, M.J., Cheng, C.P., Han, X.C., Song, B.W. (2019). Python Programming -- from Basic Development to Data Analysis, Tsinghua University Press,2019
- [21] Phillips, D. (2015). Python 3 Object-Oriented Programming. Packt Publishing Ltd.
- [22] Matthes, E. (2015). Python Crash Course: A Hands-on, Project-Based Introduction to Programming, No Starch Press.
- [23] Liu, Z.Y., Shaikh, Z., Gazizova, F. (2020). Using the Concept of Game-Based Learning in Education. International Journal of Emerging Technologies in Learning, 15(14): 53-64. <https://doi.org/10.3991/ijet.v15i14.14675>
- [24] Pei, J.Y., Shan, P. (2019). A micro-expression recognition algorithm for students in classroom learning based on convolutional neural network. Traitement du Signal, 36(6), 557-563. <https://doi.org/10.18280/ts.360611>
- [25] Duchateau, F., Lumineau, N., Aalberg, T. (2018). Impact of open and linked data on bibliographic catalogs. Ingénierie des Systèmes d'Information, 23(3-4), 57-93. <https://doi.org/10.3166/ISI.23.3-4.57-93>
- [26] Hao, J., Ho, T.K. (2019). Machine learning made easy: A review of scikit-learn package in Python programming language. Journal of Educational and Behavioral Statistics, 44(3): 348-361. <https://doi.org/10.3102/1076998619832248>
- [27] Li, J.W., Duan, Z.S., Wang, S.Y. (2018). The storage technology in NoSQL of MongoDB. Computer Knowledge and Technology, 14(27): 7-9.
- [28] Zhao, F., Lei, Z.Z., Yang, X.Y., Su, P.J., Wang, S.Y. (2018). Analysis on the network public opinion of university students. Computer Knowledge and Technology, 14(28): 227-229.

## 8 Authors

**Shunye Wang** is a teacher (Master, Associate Professor) who works in the Department of Electronic Information Engineering, Langfang Normal University. Her research interests include graph theory and its application in social network and intelligent transportation, youth artificial intelligence education and educational data mining. E-mail: [shunyew@163.com](mailto:shunyew@163.com)

**Dayong Liu** is a teacher (Ph.D., Associate Professor) who works in the Department of Electronic Information Engineering, Langfang Normal University. His research interests include Artificial Intelligence and applications, and Wireless Sensor Networks. E-mail: [1270360813@qq.com](mailto:1270360813@qq.com)

**Ning Wang** is a teacher (Master, Assistant) who works in the Department of Electronic Information Engineering, Langfang Normal University. Her research interests include computer vision and Internet of Things applications. E-mail: [wnbright567@sina.com](mailto:wnbright567@sina.com)

**Yanxiang Yuan** is a teacher (Master, Lecturer), who works in the Department of Mathematics and Statistics of Qiannan Normal University for Nationalities. Her main research interests include Differential Equations and Dynamical Systems. E-mail: [yanxiangyuan12@163.com](mailto:yanxiangyuan12@163.com)

Article submitted 2020-09-02. Resubmitted 2020-10-12. Final acceptance 2020-10-12. Final version published as submitted by the authors.