Learning Path Generation Method Based on Migration Between Concepts

Dan Liu[1,3], Libo Zhang[1,2](✉), Tiejian Luo[1], and Yanjun Wu[2]

[1] University of Chinese Academy of Sciences, Beijing 101408, China
zsmj@hotmail.com
[2] Institute of Software, Chinese Academy of Sciences, Beijing 100190, China
icode@iscas.ac.cn
[3] Nankai University, Tianjin 300350, China

Abstract. The learning strategies often have a direct impact on learning effects. Often, the learning guidance is provided by teachers or experts. With the speed of knowledge renewal going faster and faster, it has been completely unable to meet the needs of the learner due to the limitation of individual time and energy. In order to solve this problem, we propose a learning strategy generation method based on migration between concepts, in which the semantic similarity is creatively applied to measure the relevance of concepts. Moreover, the concept of jump steps is introduced in Wikipedia to measure the difficulty of different learning orders. Based on the hyperlinks in Wikipedia, we build a graph model for the target concepts, and achieve multi-target learning path generation based on the minimum spanning tree algorithm. The test datasets include the books about Computer Science in Wiley database and test sets provided by volunteers. Evaluated by expert scoring and path matching, experimental results show that more than 59% of the 860 single-target learning paths generated by our algorithm are highly recognized by teachers and students. More than 60% of the 500 multi-targets learning paths can match the standard path with 0.7 and above.

Keywords: Learning path · Wikipedia · Semantic similarity · Graph model

1 Introduction

In the era of knowledge explosion, finding an efficient way to the target knowledge, called as learning path in this paper, in these too many learning materials is a problem needed to be solved. It is necessary for learners to find a proper way grasping the knowledge they need, as it is universal that people reach their destinations efficiently relying on navigators. Traditionally, most of the learning strategies are made by teachers or experts. As a result, the learning strategies are personally and subjective. In the process of learning, learners will also have problems such as cognitive overload or cognitive impairment due to the inappropriate learning order of concepts which leads to the inefficiency of learning. Therefore, in the information age with knowledge expanding and updating rapidly, it has been far from being able to meet the needs of learners relying solely on the individual or an educational group.

The data generated by human activities are enriching the knowledge space constantly. The development of Internet has greatly accelerated the growth of information,

G. Li et al. (Eds.): KSEM 2017, LNAI 10412, pp. 495–506, 2017.
DOI: 10.1007/978-3-319-63558-3_42

which makes people begin to explore the way to organize the useful information, so there is a knowledge organization form named knowledge base which is a knowledge cluster after the artificial systematization and standardization, such as Wikipedia.

In recent years, the rapid development of the World Wide Web has led to the study of knowledge space. Lexical relations diagram, semantic relevance, knowledge ontology, semantic network, etc. have been applied to the text classification, word disambiguation, machine translation and so on. These studies also provide a theoretical basis for the learning path generation.

The knowledge base absorbs new knowledge and establishes the connection between concepts quickly while the book is authoritative and academic. Therefore, in this paper, we put forward a learning path generation method making full use of the knowledge base and traditional books to help the learners learn the target knowledge better and faster.

Our main contributions are as follows:

- We build a basic concept set in the computer domain and propose a graph model for concepts to express the concepts and their relationships;
- We propose a learning path generation method based on migration and similarity between concepts;
- We build two test data sets and evaluate the generated learning path by scoring and the consistency degree between paths.

In this paper, we creatively apply semantic similarity to the generation of learning path, and propose a new graph model for concepts to express their relationships. We use Wikipedia to construct the graph model for the target concepts automatically, without manual assistance.

Our study takes Computer Science discipline as an example, but our method's scope of application is not limited to Computer Science. Our method is based on Wikipedia and the books on the subject, which makes it possible to migrate to other subject areas at low cost, thus benefiting more learners in various fields.

The rest of this paper is organized as follows. In Sect. 2, we review related works. In Sect. 3, we propose our method of the learning path generation. Section 4 describes the validation of our results based on scoring and path matching. Finally, in Sect. 5, we give the conclusion.

2 Related Work

2.1 Approaches to Learning Path Generation

Nowadays, the speed of knowledge update is going faster and faster, and there is a high demand for learning ability and learning efficiency. The rapid development of the Internet brings people into an era of e-learning, more and more scholars turn to the automatic generation of learning strategies. The existing methods can be divided into three categories: individual based methods, knowledge level based methods, learning difficulty based methods.

Individual based methods often use learners' gender, personality, major, social background and other attributes to recommend learning paths for learners. Lin et al. [1] developed a personalized creative learning system (PCLS) in 2013 based on decision tree in data mining technology. PCLS includes a series of creative tasks and a questionnaire that involves several key volumes. It determines the level of creativity according to individual attributes and tasks. There are some one-sidedness since such methods solely rely on individual attributes, and it is not effective when learners are unwilling to provide such information.

Knowledge level based methods get the knowledge not grasped by learners according to the wrong answered questions in the pre-test. Then the learning path is recommended accordingly. Lendyuk et al. [2] proposed an individual learning path generation method using the learning object sequence to navigate the learning content. They adjusted the difficulty of problem next stage to solve according to the number of right answered questions in previous stage which allowed learners to improve their knowledge level gradually. Such methods depend on the question bank, which requires a lot of manpower to construct. Moreover, the knowledge area covered by the question bank is limited.

Learning difficulty based methods constantly adjust the learning difficulty, which is given by human initially, according to learning effect. Zhao and Wan [3] built a graph model for knowledge units. Each node represents a knowledge unit, and the directed edges represent the relationships between knowledge units: precedence, succession or parallel. The weight of the edge represents the difficulty of learning from the precedence concept to the succession concept. The initial value was given by the teacher, and then updated according to the learning of the previous students. Bonifati et al. [4] proposed a learning path generation method that satisfies the path query conditions of the user on the graph database. The input of the algorithm includes a graph database in which the user marks the node as a positive or negative case based on whether he wants the node to be part of the query result. Such methods not only consume a lot of manpower, but also have a strong subjectivity due to the artificial annotation. Furthermore, it is more and more difficult to keep up with the speed of knowledge updating for an individual even a group of experts.

In view of these problems, the method proposed in this paper combines the Wikipedia and online library resources with the consideration of the update speed and authority of the knowledge. Moreover, the migration between concepts in Wikipedia is used to measure the learning difficulty rather than artificial annotation.

2.2 Word Similarity

The word similarity discussed in this paper is at the semantic level. The methods of calculating word similarity are divided into four categories according to the basic method adopted: word similarity calculating based on vector space model, word similarity calculating based on text attributes, word similarity calculating based on sequence alignment and word similarity calculating based on semantic analysis.

In the vector space model, the text is regarded as a collection of basic language units (word, phrase, etc.), and it is assumed that the text feature is only relevant to the frequency of some basic language units, with its position or order in the text not considered completely. Word similarity calculating based on text attributes is driven by the goal of the task. It often extracts the text attributes that contribute to the certain task. At present, the text attributes used in the study include the new word occurrence rate [5], the overlap rate of words [6], the text graph model [7], etc. The text is treated as a sequence of characters when calculating word similarity based on the sequence alignment, so that the text similarity calculation turn into the calculation of sequence similarity. Commonly used methods include Hamming Distance, Edit Distance, the Longest Common Subsequence, etc. Word similarity calculating based on semantic analysis uses synonymy, antisense, upper and lower relations to calculate the similarity [8]. The advantages and disadvantages of the four kinds of methods are shown in Table 1.

Table 1. Advantages and disadvantages of the four kinds of methods.

Calculation methods	Advantages	Disadvantages
Based on the vector space model	Computational overhead is small	The effect is not good when applied to the text containing less words; Ignore the order and ambiguity of words
Based on text attributes	Focus on a certain task, more flexible and more feasible	Poor portability due to strong targeted
Based on sequence alignment	Consider the text order	Ignore the semantic relations between words
Based on semantic analysis	Discover the deep semantic relations between features	Computational overhead, high resource requirements

We used the vector space model to calculate the similarity between words via the word2vec, a tool provided by Google. We trained a CBOW (Continuous Bag-of-Words Model) model of word2vec using the Wikipedia corpus described in Sect. 3.2.

3 Generating a Learning Path

3.1 Definition of Learning Path

Learning path is a list of relevant concepts. The relevant concepts refers the concepts have semantic connection, which can be measured by word similarity. In this paper, the learning paths are divided into two kinds:

- **Single-Target Learning Path.** An orderly sequence of concepts generated for one target concept that needs to be learned. The relevant concepts are ranked in descending order of the concept's relevance to the target concept. For example, the learning path of the target concept *jQuery* can be the list: *JavaScript*, *jQuery UI*, *plugin*, *jQuery Ajax*, *HTML*, *jQuery*.

- **Multi-target Learning Path.** Learners have several concepts that need to be learned. For example, when they preview a new unit, they want to learn some emerging concepts of the unit, which are often relevant. These concepts that need to be learned constitute the target concept set, and the multi-target learning path is a directed graph generated for the target concept set, in which the node represents the target concept, and the direction of the edge represents the learning order. For example, the learning path of the target concept set {*array, tree, stack, list, queue*} can be the learning order shown in Fig. 1.

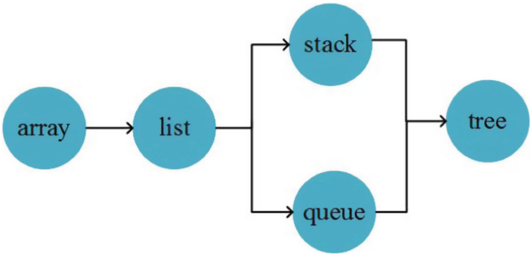

Fig. 1. A multi-target learning path.

3.2 Data Preprocessing

We propose a learning path generation method based on Wikipedia, the authoritative course guidelines and professional books.

- **Wikipedia.** The Internet Encyclopedia is the largest and most complete human knowledge base that is currently available for random access. Wikipedia is the largest Internet multilingual encyclopedia compiled by humans. As of April 8, 2017, Wikipedia has a total of 5,378,718 articles, 41,883,956 pages, which cover most of the human knowledge areas. The original data set used to build the graph model and train the similarity model is a compressed file for all Wikipedia pages (2016.06.01), a 53.4 GB XML file after decompression. In the experiment to determine the relevant concepts of the target, the online Wikipedia is used because of the consideration of web page redirection. We extracted the title of each page as an entry from the original XML file, and extract the text with a hyperlink in the page (hereinafter referred to as hyperlink text) to build data set called as wiki_R. Each line in wiki_R.txt represents the contents of an entry in the following format:

$$title\#outlinks_1\#outlinks_2\#\ldots\#outlinks_n\#$$

  Where *title* is the page's title, and $outlinks_i(\mathrm{i} = 1, 2, \ldots, \mathrm{n})$ is a hyperlink text in the page corresponding to the entry. "#" is the delimiter.
- **Course Guidelines.** Computer Science is a fast-growing discipline with much knowledge. Learners often feel confused when they study. Therefore, we choose the Computer Science as an example. In the field of computer, ACM and IEEE are the authoritative institutions, and the two agencies jointly issued 13 guidelines to lead

the direction of the development of computer courses: CC1991, CC2001, IS2002, CE2004, SE2004, CC2005, IS2006, IT2008, CS2008, SE2009, IS2010, CS2013 and SE2014, all of which are incorporated into the original data set in this study. The basic concept set volume_R can be obtained by analyzing the distinction of computer-related basic concepts in the document, where $D_i (i = 1, 2, \ldots, 13)$ represents the original guidelines:

a. Delete the URL and special characters in D_i, and then extract the fields around "()" and "/", capitalized phrases, phrases with hyphens to document d1, d2, d3, d4;
b. Filter the document d1, d2, d3, d4 artificially. Removing those words that not belong to Computer Science. Then construct R_i after duplication eliminating;
c. Repeat the above steps for each course guideline. Finally summarize R_i and eliminate duplication and then get the result document volume_R.

- **Books.** In the computer field, the knowledge update is fast. As a result, the books are too many. This study will look to the computer field part in Wiley database. Wiley, founded in the United States in 1807, is the world's largest independent academic book publisher and the third largest academic journal publisher. In this paper, we use the Wiley online collection of books subscribed by the Chinese Academy of Sciences collecting books from Wiley-Blackwell, Wiley-VCH, Jossey-Bass and so on. We download 181 books in computer field to construct a book library CSLibrary. The 181 books are divided into three categories: 75 of Computer Science, 67 of General Computing, 39 of Information Science and Technology. In order to facilitate the follow-up experiment, we use PDFMiner to turn pdf document into txt document.

3.3 Extraction of Relevant Concepts

How to find the relevant concepts for the target concept is the key to constructing our model. In this section, we extracted relevant concepts from Wikipedia and books based on word similarity discussed in Sect. 2.2.

- **Extract Relevant Concepts from Wikipedia.** In Wikipedia, the concept exists in the form of an entry, and its structure is reflected by the hyperlinks between pages. We find these links have a large or small correlation with the concept *tc*. Therefore, we use the hyperlinks to extract relevant concepts from Wikipedia. The specific steps are as follows:

a. Extract all the hyperlink texts from the Wikipedia page corresponding to the target concept *tc*. Name the hyperlink text set as HLinkSet;
b. Calculate the similarity between the target concept *tc* and each hyperlink text (concept) *rc* in the HLinkSet and rank the concepts in descending order of the *rc*'s relevance to *tc*, recorded as HLinkList;
c. Take the first *k* concepts in HLinkList to form the relevant concept set named as RCsFromWiki of the target concept *tc*.

- **Extract Relevant Concepts from Books.** A concept is closely relevant to its context. Thus, it can be assumed that the concepts appearing adjacently in the text are related to each other stronger. We try to use the computer course guidelines published by ACM and IEEE to establish a basic concept set in computer domain. This process has been described in detail in Sect. 3.2. It should be noted that the basic concept set volume_R can not completely cover the basic concepts in the book, for which we make full use of the index of the book. We extract relevant concepts from books as follows:

a. For a book *b*, extract the basic concepts that appear in the book according to volume_R and record the page number it appears in the book. Add these concepts to the set BasicSet;
b. Add the search terms that appear in the index of *b* and the page number that it appears in the book to BasicSet;
c. Rank the concepts in BasicSet according to the page number they first appear. Then we get BasicList;
d. For a target concept *tc* in the book *b* that needs to be learned, search for its index in the BasicList, denoted as *index0*. Construct the candidate relevant concept set CandidateSet with the concept in [*index0*-*m*, *index*-1] and [*index*+1, *index*+*m*] scope of BasicList. The size of the set is identified as *2m* (*2m* > *k*);
e. For each concept *rc* in CandidateSet, calculate the similarity between the target concept *tc* and *rc*, and rank it according to the similarity degree. The result is denoted as CandidateList;
f. Take the first *k* relevant concepts in CandidateList to constitute the relevant concept set RCsFromBook, where *k* is in order to keep consistent with the number of relevant concepts extracted from the Wikipedia.

3.4 The Generation Method

For the two cases, we propose different generation methods respectively.

- **Single-Target Learning Path Generation**

The single-target learning path is to help learners have a better understand of the target concept. The simple-target learning path generation method combines the relevant concepts extracted from Wikipedia and books. The specific steps are as follows:

a. For the single target concept *tc* given by the learner, extract the candidate relevant concept set from the corresponding Wikipedia page (which may be redirected pages) as RCsFromWiki (size: *k*); extract the candidate relevant concept set as RCsFromBook (size: *k*);
b. Take the union of RCsFromWiki and RCsFromBook as RCs (the size of the set is not greater than *k*). For each concept *rc* in RCs, calculate the similarity between the target concept *tc* and *rc*, and rank concepts according to the similarity to get RCsList;
c. Take the first *k* concepts in RCsList as the single-target learning path of *tc*.

- **Multi-target Learning Path Generation**

Given a concept set, how to determine the order of these concepts in order to minimize the difficulty of learning? In other words, how to reduce time and effort spent as far as possible? To solve this problem, the concept of *jumps* is introduced in Wikipedia to measure the learning difficulty between two concepts. For the two pages in Wikipedia *page1* and *page2*, the *jumps* of *page1* to *page2* are defined as:

The minimum number of clicks required clicking on the link in *page1* to finally jump to *page2*. If the number is greater than MAX_CLICK, we said that the distance of *page1* to *page2* is infinity (∞).

After introducing the concept of *jumps*, we can build a weighted directed graph model for a concept set, in which the node represents the target concept, and the direction of the edge represents the jump direction between pages corresponding to the concepts, the weight of the edge is the *jumps*. It can be seen in Fig. 2.

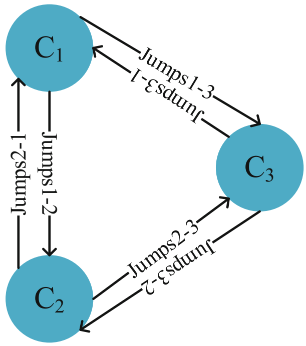

Fig. 2. Graph model based on jumps between Wikipedia pages.

With the definition of *jumps* and the graph model constructed successfully, we can get the multi-target learning path generation method:

a. Calculate the *jumps* for any of the two concepts tc_i and tc_j in the given target concept set TCs and construct the graph model accordingly;
b. Find the “path” to learn the target concepts with the least number of jumps, that is, the minimum support tree for the connected graph built in a., which is the multi-target learning path.

4 Test

We set up a series of experiments to detect the effectiveness of the proposed method.

4.1 Test Data

- **Test Data for Single-Target Learning Path Generation Method**

For the single-target learning path, we selected the book containing more than 150 basic concepts as the original test data set based on CSLibrary described in Sect. 3.2, for a total of 86 books: 60 of Computer Science, 23 of General Computing, 3 of Information Science and Technology. We randomly selected 10 basic concepts from each book to constitute the test set test_860 which contains a total of 860 basic concepts obviously.

- **Test Data for Multi-target Learning Path Generation Method**

For the multi-target learning path, there is some correlation between concepts in the targets set, that is, the learner conducts a study with a certain goal, and the relevant concepts need to be studied under the navigation of the goal. For example, When the learner wants to learn file operation while he learns C++ programing language, his target concept may be *read*, *write*, *buffer*, *seek* etc., but it seems not likely to be the *stack*, *cache*, *router* such concepts that are not consistent with the goal. Therefore, in order to ensure the test is practical enough, we use artificial data sets rather than randomly generated. We collected 500 target concept sets from the student volunteers. The target concept set size varying from 6 to 10. The ratio is 1:1:1:1:1, respectively, denoted as TCSet_6_100, TCSet_7_100, TCSet_8_100, TCSet_9_100, TCSet_10_100.

4.2 Evaluation

- **Evaluation Method for Single-Target Learning Path**

For the evaluation of the single-target learning path, we adopted the teacher and student scoring strategy:

a. Invite 50 teachers, covering lecturers, associate professors, professors, the ratio of which is 2:1:1. Invited 100 students as volunteers, whose study stage cover the undergraduate third grade, fourth grade and graduate first grade, graduate sophomore, graduate third grade, doctor first grade, the proportion of which is 1:2:2:2:2.
b. Each teacher student give a score in 0–10 for the generated learning path, and finally take the average score of all teachers and students respectively, denoted as *teacher_acore* and *student_score*.
c. Take the average of teacher_score and student_score as the comprehensive score of the learning path, denoted as *final_score*.

- **Evaluation Method for Multi-target Learning Path**

For the evaluation of the multi-target learning path, we use two methods:

a. The scoring strategy referred above;
b. Invite 50 teachers to make learning paths for the target concept sets, recorded as standard learning paths. Then map each concept to a character, so that the generated learning path and the standard learning path are mapped to two strings, the result string and the standard string. Finally, calculate the similarity between the two strings based on the Levenshtein Distance:

$$Similarity = (Max(x, y) - Levenshtein)/Max(x, y)$$

Where x is the length of the result string and y is the length of the standard string. The smaller the Levenshtein Distance, the higher the similarity of the two strings, and the more consistent the result learning path and the standard learning path, in other words, the better the results of the proposed method.

4.3 Results

For the target concepts given by the learner, we tested on data set mentioned in Sect. 4.1 and evaluated according to the evaluation methods in Sect. 4.2.

- **Single-Target Learning Path Results**

We experimented on the test set test_860. In the experiment, k was set to 5 while m was set to 8. 860 learning paths were obtained according to the method proposed in Sect. 3.4, and then we calculated *teacher_score*, *student_score* and *final_score* after getting the scores from teachers and students. The scores were divided into five intervals: [0, 2), [2, 4), [4, 6), [6, 8), [8, 10]. The number of learning paths in each interval is shown in Table 2.

Table 2. The number of learning paths in each score interval.

	[0, 2)	[2, 4)	[4, 6)	[6, 8)	[8, 10]
Teachers	24	141	193	256	246
Students	43	102	208	230	277
Final	32	123	201	248	256

We can see that 96% of the 860 learning paths' *final_score* are not less than 2 points and 59% are not less than 6 points. 30% fall in the [8, 10] interval. It can be said that the generated learning paths are highly recognized by the teachers and students. However, there are scores less than 4 points. It seems led by the relativity of the relevance concepts extraction. We extract relevant concepts via hyperlinks in Wikipedia and the context of the target concept in the book. Although we filter out of the concepts with lower similarity with the target concept, the compare is too relative in some cases.

- **Multi-target Learning Path Results**

In the experiment, the MAX_CLICK was set to 6. The experimental results on the 6 test data sets mentioned in Sect. 4.1 were denoted as ResultLP_6_100, ResultLP_7_100, ResultLP_8_100, ResultLP_9_100, ResultLP_10_100.

a. Evaluation Results Using Scoring Strategy

The number of learning paths in each score interval for each of the six result sets is shown in Table 3.

From Table 3 we can see that even if the size of the target concept set is different, the difference of learning paths in each score interval is not more than ±10, and the number of generated learning paths gaining score not less than 6 is more than 62. We can draw the conclusion: The size of the target concept set has less influence on the results. Moreover, more than 62% of the generated learning paths gain a score not less than 6.

Table 3. The number of learning paths in each score interval when the target concept set size varying from 6 to 10.

	6	7	8	9	10
[0, 2)	2	3	2	1	2
[2, 4)	12	10	12	11	9
[4, 6)	23	25	24	23	21
[6, 8)	30	28	25	27	26
[8, 10]	33	34	37	38	42

When building the graph model for the target concept set, the value of *jumps* between two pages depends on the links in the page totally. Due to indeterminacy of the page links and the editors' subjectivity, there will be the case that the value of *jumps* is greater than MAX_CLICK. More generally, there may be the case that the value of *jumps* from concept c_1 to c_2 is equal to the value of *jumps* from concept c_2 to c_1, and the generated learning path will have a random sequence in c_1, c_2 which will lead to the deviation.

b. Evaluation Results of Standard Path Comparison

We invited 50 teachers to make learning paths for TCSet_6_100, TCSet_7_100, TCSet_8_100, TCSet_9_100, TCSet_10_100 as standard learning paths. In accordance with Sect. 4.2, the generated learning path and the standard learning path on the same target concept set were mapped to two strings, and we calculated their similarity to measure the consistency between them. The path consistency with the target concept set size varying from 6 to 10 is shown in Table 4.

Table 4. Path consistency with the target concept set varying from 6 to 10.

	6	7	8	9	10
[0, 0.5)	3	5	4	6	5
[0.5, 0.7)	31	31	36	29	24
[0.7, 1.0)	66	64	60	65	71

It is shown that when the size of the target concept set is 6, 7, 8, 9 or 10, the difference of the number of learning paths falling in each similarity interval is less than ±5, and the number of learning paths with the consistency not less than 0.7 compared to the standard learning path was more than 60. We can draw the conclusion: The size of the target concept set has less influence on the results. Moreover, more than 60% of the generated learning paths can match the standard path with 0.7 and above.

5 Conclusions

In this paper, we focus on the problem how to help learners to learn the knowledge they need better and faster. In view of the limitations of the existing learning path generation methods, we propose a learning path generation method with the combination of the traditional learning and modern network learning aiming at all learners.

We creatively apply semantic similarity to the generation of learning strategies to measure the relevance of concepts and introduce jump steps in Wikipedia to measure the difficulty of different learning orders. Based on the hyperlinks in Wikipedia, the graph model can be built successfully for the target concepts, which is a key step in multi-target learning path generation. We test the proposed method on the books about Computer Science in Wiley database and test sets provided by volunteers. The expert scoring results show that more than 59% of the 860 single-target learning paths generated by our method are highly recognized by teachers and students and more than 62% of the 500 multi-target learning paths gain a score not less than 6. By path matching, it can be seen that more than 60% of the 500 multi-target learning paths can match the standard path given by experts with 0.7 and above.

Considering the truth that the graph model in this paper is built automatically, without any manual work, our method is manpower saving and the results are more objective. In future research, a broader library of books and a basic concept set covering more fields will be built to serve more learners.

References

1. Lin, C.F., Yeh, Y.C., Hung, Y.H., Chang, R.I.: Data mining for providing a personalized learning path in creativity: an application of decision trees. Comput. Educ. **68**, 199–210 (2013)
2. Lendyuk, T., Melnyk, A., Rippa, S., Golyash, I., Shandruk, S.: Individual learning path building on knowledge-based approach. In: 8th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), pp. 949–954. IEEE, Warsaw (2015)
3. Zhao, C., Wan, L.: A shortest learning path selection algorithm in e-learning. In: 6th International Conference on Advanced Learning Technologies, pp. 94–95. IEEE, Kerkrade (2006)
4. Bonifati, A., Ciucanu, R., Lemay, A.: Learning path queries on graph databases. In: 18th International Conference on Extending Database Technology (EDBT), Brussels (2015)
5. Youmans, G.: A new tool for discourse analysis: the vocabulary-management profile. Language, 763–789 (1991)
6. Fragkou, P., Petridis, V., Kehagias, A.: A dynamic programming algorithm for linear text segmentation. J. Intell. Inf. Syst. **23**(2), 179–197 (2004)
7. Malioutov, I., Barzilay, R.: Minimum cut model for spoken lecture segmentation. In: 21st International Conference on Computational Linguistics, pp. 25–32. Association for Computational Linguistics, Sydney (2006)
8. Luo, T., Zhang, L., Yang, L., Chen, X.: TACE: a toolkit for analyzing concept evolution in computing curricula. In: 28th International Conference on Software Engineering and Knowledge Engineering, SEKE, Redwood City (2016)