Towards Automatic Building of Learning Pathways

Patrick Siehndel[1], Ricardo Kawase[1], Bernardo Pereira Nunes[2] and Eelco Herder[1]
[1] *L3S Research Center, Leibniz University Hannover, Hannover, Germany*
[2] *Department of Informatics, PUC-Rio, Rio de Janeiro, RJ, Brazil*

Keywords: Learning Support, Learning Pathways, Digital Libraries.

Abstract: Learning material usually has a logical structure, with a beginning and an end, and lectures or sections that build upon one another. However, in informal Web-based learning this may not be the case. In this paper, we present a method for automatically calculating a tentative order in which objects should be learned based on the estimated complexity of their contents. Thus, the proposed method is based on a process that enriches textual objects with links to Wikipedia articles, which are used to calculate a complexity score for each object. We evaluated our method with two different datasets: Wikipedia articles and online learning courses. For Wikipedia data we achieved correlations between the ground truth and the predicted order of up to 0.57 while for subtopics inside the online learning courses we achieved correlations of 0.793.

1 INTRODUCTION

When learning about a new topic, especially in a domain that is new to the learner, it is not always directly clear in which order relevant resources can best be read or learned, ensuring that the basic concepts are introduced first, followed by more advanced material that elaborates on these concepts. This is commonly known as *Learning pathway*. In fact, a learning pathway is described as the chosen route, taken by a learner through a range of learning activities, which allows them to build knowledge progressively(Jih, 1996).

Our approach exploits latent concepts inside learning objects and, according to the estimated complexity of these concepts, provides a tentative ordering for a set of learning objects. The results provide learners with an ordered learning script to follow, similar to a course in which lectures are arranged in a specific order.

For our method, we exploit information from Wikipedia, which we use as an external knowledge base. Wikipedia contains over 4 million articles (concepts) that virtually cover all concepts that are relevant for referencing. Further, each Wikipedia article contains links to reference articles and it is manually categorized. We exploit a set of features extracted from Wikipedia and its category graph to estimate the complexity of a given text. Our methods are based on the assumptions that:

- Wikipedia categories contain a useful link structure for ordering objects based on their difficulty;
- Concepts that are mentioned inside Wikipedia articles provide useful background knowledge for understanding the meaning of an article.

Our method uses the Wikipedia Miner[1] toolkit for detecting concepts in the analyzed learning objects. The detected concepts are basically text snippets that can be related to a Wikipedia article. All Wikipedia articles belong to one or more categories, and these categories are organized in a graph structure. We use this graph structure for identifying categories that are more general and therefore supposedly known by a user.

The main aspect of our work is to help learners to identify a meaningful order of given learning material. An example: in mathematics, it is obvious that learning basic principles like summing or dividing should come before starting with topics such as ‘curve sketching’. Essentially, the problem we aim to solve can be summarized as follows: given a set of learning objects, we bring them into a reasonable order, to help learners finding a good starting point as well as a good way through the provided material.

The rest of the paper is organized as follows: In Section 2, we discuss related work on the topics of learning object recommendation and ordering. The proposed method is explained in detail in Section 3.

[1] http://wikipedia-miner.cms.waikato.ac.nz/

Siehndel P., Kawase R., Pereira Nunes B. and Herder E..
Towards Automatic Building of Learning Pathways.
DOI: 10.5220/0004837602700277
In *Proceedings of the 10th International Conference on Web Information Systems and Technologies* (WEBIST-2014), pages 270-277
ISBN: 978-989-758-024-6

The experimental evaluation of the whole process is presented in Section 4, where we used two different data sets to analyze the performance of our method: Wikipedia articles and online learning courses from Khan Academy. We conclude the paper in Section 5 by summarizing our main contributions.

2 RELATED WORK

A dynamic course generator is presented by Farrell et al. (Farrell et al., 2004). The course is assembled based on keyword queries and the metadata of learning objects contained in a given repository. The sequence relies on the relationships that are manually assigned to each learning object and its classification (e.g. introduction, methodology or conclusion). Hence, the objects are selected and reordered according to a user query and its classification. Chen (Chen, 2008) present an evolutionary approach that uses a genetic algorithm to generate a learning plan. The genetic approach is based on a pretest performed by students, where missed concepts help in creating new learning plans, according to concepts and levels of difficulty of the learning objects. Our approach follows the dynamic nature of these approaches, since we only need an input concept to determine a learning object sequence.

Ullrich and Melis (Ullrich and Melis, 2009) order learning objects according to the learning goal of each student. For this, they classify objects into different classes, such as *illustrate* or *discover*, where the course is assembled by a sequence of examples or in depth.

In the areas of Intelligent Tutoring Systems and Adaptive Hypermedia the adaptive sequencing is common technique (Brusilovsky and Millán, 2007). In scenarios where metadata for the given learning objects is available systems like PASER (Kontopoulos et al., 2008) allow the calculation of a learning path. In our scenario we address informal learning situations in which this metadata is not available.

Another perspective on the sequencing of learning objects is discussed by Kickmeier-Rust et al. (Kickmeier-Rust et al., 2011), where they use a combination of a storytelling model and competence structures to identify the learning state of a student in games. By identifying the state of the student, they propose a new sequence of learning objects, while keeping the story lines. Limongelli et al. (Limongelli et al., 2009) present a framework to create personalized courses. The sequencing of learning objects is generated taking into account the cognitive state of the student and her learning style. Sequences change according to the results obtained by the students, in order to cover a concept not understood. Missed concepts are identified through exercises during the learning process. Instead of discovering the learning state of students, we focus on a general applicable approach. Our approach identifies which topics are necessary to understand a topic independent of the student. On the one hand, we do not provide personalized learning paths; on the other hand, we overcome the cold-start problem where there is no a priori information of the students.

Champaign and Cohen (Champaign and Cohen, 2010) introduce a work based on student development after consuming a given learning object. Each student is assessed and the most successful sequence of learning objects is selected and recommended to students with similar profiles. Similarly, Knauf et al. (Knauf et al., 2010) focus on similar profiles to recommend similar learning paths. However, the similarity between students is based on learning models that describe the abilities of each student. A path taken by a successful student is recommended to another one with similar characteristics. In contrast, the goal of our approach is to recommend learning objects following the learning goals of a student; as the student selected a topic to learn, the sequencing is determined by knowledge and concepts needed to understand a learning object.

3 METHOD

Our method for ordering learning objects and providing background links is divided into two main steps. The first step is the annotation of the content with links to relevant Wikipedia articles. This step is described in more detail in Section 3.1. The second step exploits detected topics and Wikipedia as a knowledge base to calculate the order of learning objects in a given set.

3.1 Annotation and Features

For annotating the content of the given learning objects, we used the Wikipedia-Miner Toolkit (Milne and Witten, 2008). The tool annotates a text (links terms to articles) in the same way a human would do it in Wikipedia. With this information, based on the detected topics inside a given learning object, we calculate a set of features that indicate the complexity and relevance of a topic. The features we use for ordering the given objects are:

1. *Number of inlinks:* the number of Wikipedia articles that link to the detected topics.

2. *Number of outlinks:* the number of links to other articles contained by detected topics.
3. *Text Length of Linked Articles:* the length of the detected articles.
4. *Average Word Length of Linked Articles:* the average length of the words in the detected articles.
5. *Average Word Length of Learning Object:* the average length of the words in the learning object.
6. *Distance to Root of Linked Articles:* the average distance to the root categories of the articles.
7. *TF/IDF Score of Words in Linked Articles:* the TF/IDF values of the words inside the detected articles.
8. *TF/IDF Score of Words in Learning Object:* the TF/IDF values of the words inside the learning object.

The first two features are chosen based on the assumption that the number of inlinks and outlinks are indicators of the generality of a Wikipedia article: if many articles link to one page, it indicates that this concept is a basic (popular) concept. As in (Kamps and Koolen, 2009), inlinks and outlinks are deemed to be good indicators of an article's relevance to a given topic.

We also assume that the text length and the average words length are good indicators about how complex a topic is. Another important feature is the average distance of the related categories to the root node of the category tree. This feature is based on the assumption that more complex topics inside Wikipedia are deeper down in the category graph and is comparable to the generality feature in (Milne and Witten, 2012). The TF/IDF feature represents the assumption that words that rarely appear inside our corpus are related to more complex topics. All of our features are represented in four ways: we use the minimum, maximum, mean and standard deviation of each of these features to represent one learning object, which gives us a 32-dimensional float vector representing one learning object.

3.2 Learning to Order Objects

Our ordering approach is based on machine learning algorithms. The given features are used to generate a model that calculates a score for every learning object. This score indicates the estimated complexity of the concepts within the learning object. In our experiments, we used four different machine learning algorithms to produce the models. Two of these algorithms create a tree structure based on the given training data. In addition, we used a regression model and a Support Vector Machine for regression to calculate the order of the given objects.

Note that the score is based on a comparison of learning objects, and this only makes sense if the learning objects cover related topics from a single domain. For example, answering the question if one should learn a topic like 'European History' before learning 'Linear functions' is out of the scope of this paper. Due to the different nature of different learning domains, the quality of the generated order is higher when only a single domain is considered. Our approach can help users to decide which object in a given set might be useful to be learned first, assuming that the objects are related per se.

4 EXPERIMENTAL EVALUATION

In this section we evaluate the performance of the proposed method by analyzing the quality of the predicted order of different sets of learning objects. We performed our evaluations with two different datasets: Wikipedia articles from different domains and a large set of online learning courses from the Khan Academy[2]. We chose these datasets, as they contain elements that can be used as a ground truth that indicates how complex a given element is. For the Wikipedia articles, we chose the distance from the root node as an indicator for complexity. For the online course dataset, we exploited its hierarchical structure, which also indicates an order in which the elements should be learned.

4.1 Ordering Evaluation with Wikipedia Data

In this section, we describe the outcomes of our experiments with Wikipedia data. We show that there are useful correlations between the depth of a concept in the Wikipedia tree and other features that we use to define the complexity of a topic. For Wikipedia articles there is no predefined order that defines which article one should read first. We decided to take the distance to the root node of an article as an indicator for the complexity of the given topic. Every Wikipedia article belongs to at least one category, and based on the conventions how articles are added to categories, the articles should be added to the most specific category. Due to this, articles that belong to lower level categories cover in most cases more specific topics.

[2] https://www.khanacademy.org/

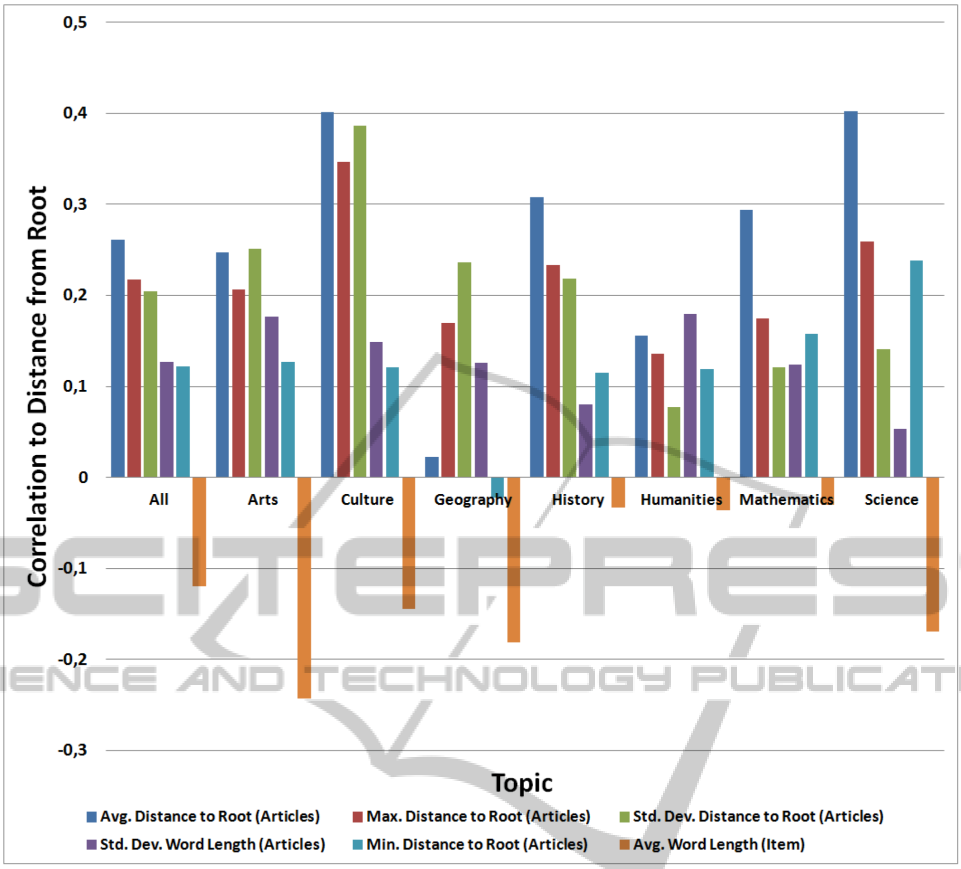

Figure 1: Correlation between features and distance to root of Wikipedia articles.

Table 1: Average distance to the root of articles from different categories.

Category	Arts	Culture	Geography	History
Avg. Distance	4.329	5.468	5.726	4.436
Category	Humanities	Mathematics	Science	Average
Avg. Distance	4.869	3.321	4.1	4.06

4.1.1 Dataset

In order to understand how the different features correlate and to get a first overview, we analyzed sets containing 500 articles from different Wikipedia Main Topics. The main topic is defined by following the category graph up to the first level of categories. The categories we chose to analyze are: 'Arts', 'Culture', 'Geography', 'History', 'Humanities', 'Mathematics' and 'Science'. All of the mentioned categories also have a relation to topics taught in school and are therefore of special interest.

4.1.2 Ordering Wikipedia Articles

For learning an order inside the Wikipedia articles, we started by analyzing the distance to the root of articles belonging to different categories. As results show in Table 1, different categories have different average distances to the root node. This is caused by the singular link and category structures inside the different categories. In comparison to 'Mathematics', which seems to have a relative flat category graph, we see that the average distance to the root for 'Geography' articles is much higher. Due to the large differences between the different categories, we decided to also analyze the correlations between the distance to the root and the calculated features for each category separately. The results for 6 of the features we analyzed is shown in Figure 1.

Overall we analyzed the correlations between 33 different features gathered from an article and its distance from the root of the category graph. Since our primary goal is to calculate an optimal order in which items should be learned, we analyzed how our fea-

Table 2: Results of predicted distance to root for Wikipedia articles using Machine Learning Algorithms.

	SMOReg	M5P	Additive Regression	Bagging
All Articles	0.4878	0.5004	0.4422	**0.5054**
Arts	0.3587	**0.4019**	0.3611	0.3836
Culture	**0.5253**	0.509	0.5076	0.5213
Geography	0.0502	0.0027	0.3591	**0.3835**
History	**0.2819**	0.2516	0.267	0.2056
Humanities	0.0076	0.213	0.1777	**0.2373**
Mathematics	0.0907	0.4225	0.306	**0.4313**
Science	0.074	**0.5704**	0.5309	0.5478

tures correlate with the complexity of the Wikipedia articles. We divided the articles in two groups, based on their positions inside the category graph of Wikipedia. The first group consists of basic articles (distance<4), while the second group consists of advanced articles (distance>=4). Figure 1 shows the correlations between these groups and six features. The singularities between the different categories indicate that learning objects of each category may require different strategies.

It is noteworthy to mention that the feature "Distance to Root (Article)" is the most important feature. This feature is calculated based on the links to other articles inside the article that we want to rank. The positive correlations of maximum, minimum, and average show that articles that are already deep inside the Wikipedia category graph tend to have links to articles that are also deep inside this graph.

Another noteworthy fact is that the average word length inside the ranked articles has a negative correlation with the group index. This indicates that longer words (on average) tend to be in articles that are higher in the tree; this was not expected, as we expected to find longer words in articles that are deeper in the category tree.

The correlations found between the distance to the root of an article and the several features that we extracted from the articles indicate that it is possible to calculate an order for learning objects. To further analyze how well these features can be used to predict the complexity of a given text, we used machine learning algorithms to predict the actual distance to the root of a given article based on all the extracted features. The algorithms used are all integrated in Weka3[3](Hall et al., 2009). We used SMOreg (Shevade et al., 2000), which is as implementation of a Support Vector Machine for regression, M5P (Quinlan, 1992)(Wang and Witten, 1997), which implements algorithms for creating M5 Model trees and rules, AdditiveRegression (Friedman and (y X)-values, 1999), which is an improved regression-based classifier, and a Bagging-Algorithm (Breiman, 1996) based on a RepTree algorithm.

We predicted the distance for every category on its own and for all articles of the different categories together. The results are based on a 10-fold cross validation and shown in Table 2. We see that, for different categories, different algorithms produce the best result. On average, the Bagging-based approach produced the best results. Additionally, this algorithm shows a very low standard deviation over the different categories. In general, we see that the distance to the root for articles of the topics 'Arts', 'Humanities' and 'Geography' is harder to predict than in the cases of articles from 'Science', 'Mathematics' and 'Culture'.

In summary, our first set of experiments shows that it is possible to predict a meaningful order for Wikipedia articles based on features extracted from Wikipedia's link structure and the textual features within these articles.

4.2 Evaluation with Online Learning Data

In addition to the evaluation on Wikipedia Data, we performed an analysis of the proposed method on a real world dataset of learning courses. While the outcomes of the first experiments proved that the assumption that features gathered from text snippets and related Wikipedia articles can be used to calculate the complexity of given texts (by means of the distance of the article to the root node), we now use the given order of a set of learning objects as ground truth.

4.2.1 Dataset

The dataset used for this series of experiments was extracted from the online courses of Kahn Academy [4]. We analyzed the text of 2508 different lectures related to the main topics 'Math', 'Science' and 'Humanities'. These items are organized in a three-level hierarchy: the first level is a general category like 'Sci-

[3] http://www.cs.waikato.ac.nz/ml/weka/

[4] https://www.khanacademy.org/

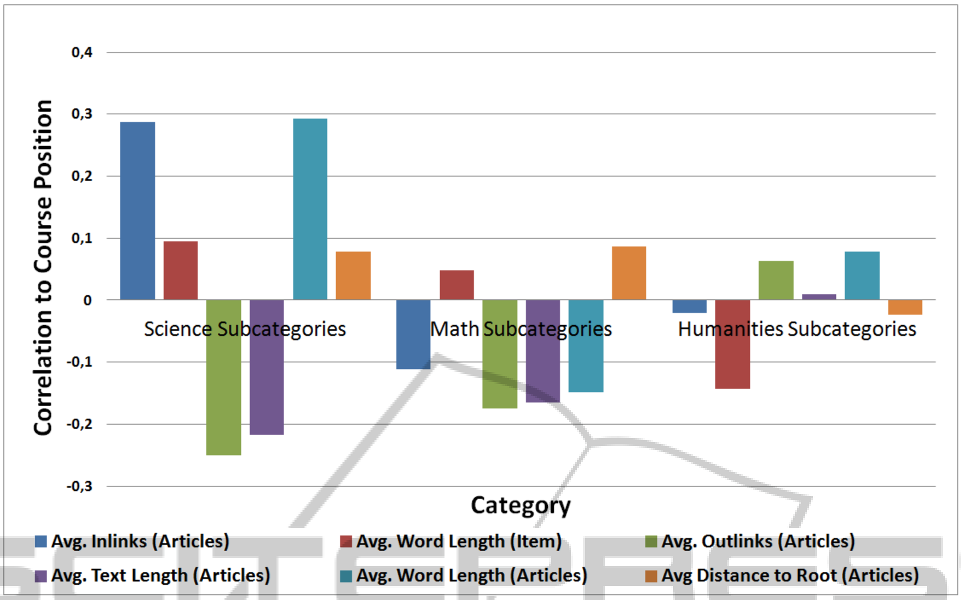

Figure 2: Correlation between features and learning object locations inside a course.

ence'; inside this category, there are different groups like 'Chemistry' or 'Biology'. Below this level are the actual courses, like 'Cell division' or 'Oxidation reduction'. The learning objects within a course are manually ordered in a meaningful way, representing the order in which a student is supposed to learn. Overall, we analyzed 110 different learning courses. Statistics on these courses are shown in Table 3

We chose to take the given order of the objects inside a course as ground truth for evaluating our approach. Calculating an order for a higher level does not make sense for all the given objects. For example, it is hard to say that 'Biology' should be learned before 'Chemistry', or that the 'Industrial Revolution' has to be learned before 'Art History', but when learning about matrices it seems to be useful to learn 'Matrix multiplication' before 'Determinant calculation'.

4.2.2 Ordering Learning Objects

We started the analysis of learning objects in the same way as we did for Wikipedia articles: by analyzing correlations between the order of objects and the calculated features. The results showed us that the order of these items follows a more complex structure that is hard to grasp by just taking into account the linear relations between the order of the objects and the values of the calculated features.

Figure 2 displays the correlation values between learning objects and different features. We can see that with the shown features we do not obtain the same correlations for all kind of topics as we got for the Wikipedia articles. A closer look at the annotated articles revealed that this is most likely caused by noise inside the transcripts of the online courses. This noise origins from the fact that the transcripts only represent the spoken content of the video lectures, which is hard to understand without the whole content of the video. Combined with a fair number of non-relevant remarks that were still included in the transcript, the quality of the extracted articles is not as good as in the previous experiment. Despite this drawback, for many of the features there are clear relations between the features and the location inside the course. We decided to perform the same tests as before to calculate the actual position of the learning objects inside the courses.

The results of this series of experiments are displayed in Table 4. The results were produced using all mentioned features using a 10-fold cross validation.

The highest overall achieved correlation between the actual position and the calculated position was at 0.554, when the algorithm is applied on all available learning objects. When training and testing on subcategories of the data, we achieve results of up to 0.793. The results differ strongly between the different domains of the online courses. For the elements of the domain 'Humanities' none of the tested algorithms achieved good results, while the order of 'Science'-related elements was relatively well calculated by all algorithms. We also see that not all different algorithms are in the same way suitable for predicting the actual rank of the items. On average, the best results were achieved using the Bagging approach.

Table 3: Statistics on the Learning Object Dataset.

Main Category	#Groups	#Courses	Avg. Items per Course
Humanities	2	18	30.92
Mathematics	5	40	33.38
Science	4	47	10.76

Table 4: Results of predicted positions of learning objects using Machine Learning Algorithms.

	SMOReg	M5P	Additive Regression	Bagging
All LOs	0.292	0.338	0.408	**0.554**
Mathematics	0.094	0.365	0.397	**0.416**
Science	0.357	0.779	0.71	**0.793**
Humanities	0.056	**0.141**	0.135	0.127
Wikipedia	0.488	0.500	0.442	**0.505**

4.3 Discussion

The series of conducted experiments shows that the proposed method can be used for calculating the complexity of a given topic, based on text features and features extracted from Wikipedia. Additionally, there are evidences that for some categories it is harder to predict its complexity than for others. Especially content from the area of Humanities seems to be harder to order than content from disciplines like Mathematics or Science. This might be due to a more complex structure of the underlying content: in Mathematics, the order in which elements need to be learned is much clearer, due to the fact that concepts build up on one another. By contrast, in disciplines like History, this is in most cases not true.

5 CONCLUSION

In this paper, we presented a method for ordering learning objects based on the complexity of the covered content. The proposed method is based on features that are extracted from the original items, as well as from the knowledge stored in Wikipedia. By using Wikipedia, we exploit a knowledge base that is constantly updated and freely available. We analyzed the performance of the method on two different datasets, and achieved correlations between the ground truth and the predicted values of up to 0.793 for special topics of learning courses. The results show that text-based learning material can automatically be sorted in a meaningful order. However, the quality varies, depending on the domain and the textual quality of the elements. For example, written text from Wikipedia is easier to order than noisy video transcripts.

The results of the experiments also showed that the proposed method works better with domains like Mathematics or Science compared to domains like Humanities or History. In general it seems to be useful to train different models for different domains since the values of some features vary over different domains.

The proposed order, as provided by our method, can help learners to find a good starting point for their learning pathways inside a set of learning resources. Also, it might help them to choose how to continue their learning process once a lesson has been learned or a resource has been visited. In addition, the methods may help teachers to analyze how the complexity of their courses evolves over time, which may help them to find a more suitable order for the elements they are teaching. A big advantage of the proposed method is that no metadata is required for calculating an order. This allows to incorporate every kind of textual resource into the learning process.

As future work we plan to build a model that can identify prerequisite knowledge for given learning courses. This will allow teachers and learners to better build a background knowledge for teaching/learning activities.

ACKNOWLEDGEMENT

This work has been partially supported by the European Commission under ARCOMEM (ICT 270239) and QualiMaster (ICT 619525).

REFERENCES

Breiman, L. (1996). Bagging predictors. *Machine learning*, 24(2):123–140.

Brusilovsky, P. and Millán, E. (2007). User models for adaptive hypermedia and adaptive educational systems. In *The adaptive web*, pages 3–53. Springer-Verlag.

Champaign, J. and Cohen, R. (2010). A model for content sequencing in intelligent tutoring systems based on the ecological approach and its validation through simulated students. In Guesgen, H. W. and Murray, R. C., editors, *FLAIRS Conference*. AAAI Press.

Chen, C.-M. (2008). Intelligent web-based learning system with personalized learning path guidance. *Computers & Education*, 51(2):787 – 814.

Farrell, R. G., Liburd, S. D., and Thomas, J. C. (2004). Dynamic assembly of learning objects. In *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, WWW Alt. '04, pages 162–169, New York, NY, USA. ACM.

Friedman, J. H. and (y X)-values, O. K. (1999). Stochastic gradient boosting. *Computational Statistics and Data Analysis*, 38:367–378.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. (2009). The WEKA data mining software: an update. *Special Interest Group on Knowledge Discovery and Data Mining Explorer Newsletter*, 11(1):10–18.

Jih, H. J. (1996). The impact of learners' pathways on learning performance in multimedia computer aided learning. *J. Netw. Comput. Appl.*, 19(4):367–380.

Kamps, J. and Koolen, M. (2009). Is wikipedia link structure different? In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, WSDM '09, pages 232–241, New York, NY, USA. ACM.

Kickmeier-Rust, M., Augustin, T., and Albert, D. (2011). Personalized storytelling for educational computer games. In Ma, M., Fradinho Oliveira, M., and Madeiras Pereira, J., editors, *Serious Games Development and Applications*, volume 6944 of *Lecture Notes in Computer Science*, pages 13–22. Springer Berlin Heidelberg.

Knauf, R., Sakurai, Y., Takada, K., and Tsuruta, S. (2010). Personalizing learning processes by data mining. In *Advanced Learning Technologies (ICALT), 2010 IEEE 10th International Conference on*, pages 488 –492.

Kontopoulos, E., Vrakas, D., Kokkoras, F., Bassiliades, N., and Vlahavas, I. (2008). An ontology-based planning system for e-course generation. *Expert Systems with Applications*, 35(1):398–406.

Limongelli, C., Sciarrone, F., Temperini, M., and Vaste, G. (2009). Adaptive learning with the ls-plan system: A field evaluation. *Learning Technologies, IEEE Transactions on*, 2(3):203 –215.

Milne, D. and Witten, I. H. (2008). Learning to link with wikipedia. In *Proceedings of the 17th ACM conference on Information and knowledge management*, CIKM '08, pages 509–518, New York, NY, USA. ACM.

Milne, D. and Witten, I. H. (2012). An open-source toolkit for mining wikipedia. *Artificial Intelligence*.

Quinlan, J. R. (1992). Learning with continuous classes. In *Proceedings of the 5th Australian joint Conference on Artificial Intelligence*, volume 92, pages 343–348. Singapore.

Shevade, S. K., Keerthi, S. S., Bhattacharyya, C., and Murthy, K. R. K. (2000). Improvements to the SMO algorithm for SVM regression. *Neural Networks, IEEE Transactions on*, 11(5):1188–1193.

Ullrich, C. and Melis, E. (2009). Pedagogically founded courseware generation based on htn-planning. *Expert Systems with Applications*, 36(5):9319 – 9332.

Wang, Y. and Witten, I. H. (1997). Inducing model trees for continuous classes. In *Poster Papers of the 9th European Conference on Machine Learning (ECML 97)*, pages 128–137. Prague, Czech Republic.