INTERACTIVE LEARNING ENVIRONMENTS
https://doi.org/10.1080/10494820.2019.1680391

Routledge
Taylor & Francis Group

Check for updates

Using learning analytics to support students' engineering design: the angle of prediction

Wanli Xing [a,b], Bo Pei[a,b], Shan Li[c], Guanhua Chen[b,d] and Charles Xie[b,d]

[a]School of Teaching & Learning, University of Florida, Gainesville, FL, USA; [b]Learning Genome Collaborative, Natick, MA, USA; [c]Department of Educational and Counselling Psychology, McGill University, Montreal, Canada; [d]The Concord Consortium, Concord, MA, USA

ABSTRACT

Engineering design plays an important role in education. However, due to its open nature and complexity, providing timely support to students has been challenging using the traditional assessment methods. This study takes an initial step to employ learning analytics to build performance prediction models to help struggling students. It allows instructors to offer in-time intervention and support for these at-risk students. Specifically, we develop a task model to characterize the engineering design process so that the data features can be associated with the abstract engineering design phases. A two-stage feature selection method is proposed to address the data sparsity and high dimensionality problems. Then, instead of relying on probability-based algorithms such as Bayesian Networks to represent the task model, this study used the Radial Basis Function based Support Vector Machines for prediction to identify the struggling students. Next, we employ an extra-tree classification method to rank the importance of these features. Teachers can integrate the feature importance ranking with the abstract task model to diagnose students' problems for scaffolding design. The results show that the proposed approach can outperform the baseline models as well as providing actionable insights for teachers to provide personalized and timely feedback to students. Implications of this study for research and practice are then discussed.

ARTICLE HISTORY
Received 31 October 2018
Accepted 10 October 2019

KEYWORDS
Learning analytics; educational data mining; engineering design; predictive modeling; feature selection; algorithms

Introduction

Engineering design has been considered a significant component of curricula in numerous countries since the 1970s (Timms, Moyle, Weldon, & Mitchell, 2018). Engineering design has been extensively incorporated into Next Generation Science Standards for U.S. science education (National Assessment Governing Board, 2013) and the Technology and Engineering Literacy Framework for national assessment (Leu, Kinzer, Coiro, Castek, & Henry, 2017). However, learning engineering design skills has always been challenging for students. Unlike professional engineers, students have yet to develop the abstract mental models and design thinking skills that engineering design requires (Atman et al., 2007). They therefore must have frequent support and scaffolding if they are to solve engineering design problems. But students rarely receive such supports because of the enormous effort it requires of teachers (Zhou, Hu, & Xu, 2018).

Researchers and practitioners have developed various ways for assessing students engaging in engineering design to support their design process, such as verbal protocol analysis, video analysis, and design product analysis (Dym, Agogino, Eris, Frey, & Leifer, 2005; Schwarz et al., 2009). These

CONTACT Wanli Xing wanli.xing@coe.ufl.edu
© 2019 Informa UK Limited, trading as Taylor & Francis Group

assessing methods not only require time-consuming data collection and analysis but also take place in a post-hoc manner. Therefore, such assessment methods are unable to provide help during the engineering design process. Given that engineering design generally involves open-ended problems with a large solution space (Xie, Zhang, Nourian, Pallant, & Hazzard, 2014) and classes may be quite large, identifying the students needing support and providing early intervention becomes almost impossible for teachers.

The emerging fields of learning analytics (LA) and educational data mining (EDM) offer promise in enabling timely feedback to students while they are engaged in the engineering design. The reason is that learning analytics and educational data mining can analyze the automatically collected digital trace data when students interact with the design platform. From this low-level trace data, it is possible to build prediction models for teachers to monitor students' progress and identify at-risk students in real time (Peña-Ayala, 2014). The automatic nature of methods using LA and EDM mean that they have the potential to reduce teachers' workload and allow them to identify students in need of help even if learning takes place online and on a large scale(Xing, Guo, Petakovic, & Goggins, 2015; Xing, Chen, Stein, & Marcinkowski, 2016). At the same time, teachers can proactively provide help to students during the design process rather than wait until they finish the design task.

While the predictive modeling of LA and EDM has been widely used in various contexts of education (e.g. online courses, collaborative learning, and intelligent tutoring systems, Baker & Gowda, 2010; Dutt, Ismail, & Herawan, 2017; Gašević, Dawson, & Siemens, 2015), to the best of our knowledge to date have explored the construction of prediction and assessment models for learning engineering design. Compared with the trace data generated in other learning contexts, the number of behaviors (variables) involved in the engineering design process can be much larger (Wu, Zhu, Wu, & Ding, 2014). That is, the data has high dimensionality. In addition, due to the open nature of design, a student may prefer one behavior to another to accomplish the design goal. The clickstreams of different students have data sparsity. That is, some behaviors are with a huge number of clicks and others might 0 or several clicks (Wu et al., 2014).

Data sparsity and dimensionality are known to influence the performance of prediction models (Szegedy et al., 2015). Without a reliable prediction model, it is impossible to provide valid support and feedback to help struggling students. This is because the diagnostic of students' learning using these prediction models are simply incorrect. Data sparsity and high dimensionality will result in a very complex task model using probabilistic algorithms (e.g. Bayesian Network) because the large number of features in the model (Segedy, Biswas, & Sulcer, 2014). Teachers will have a difficult time to associate the large number of behavioral features with the engineering design process and in turn to design relative pedagogical feedback.

In this study, we propose a methodological workflow shown as Figure 1 that aims to accurately identify students who are struggling with learning engineering design at such a time that teachers can provide timely support to them. Specifically, based on Gero's function-behavioral-structure design (FBS) model (Gero & Kannengiesser, 2014) and Holmes and Yazdani's (1999) sequential engineering design model, we created a task model (Segedy et al., 2014) of the engineering design to relate students' behavioral features with the abstract engineering design process. This task model will enable teachers to understand how students' behavioral features relates to their success or failure in completing a design task. A two-stage feature selection method was designed and implemented to deal with to deal with data sparsity and high dimensionality.

Then a specific genre of support vector machines, Radial Basis Function Based Support Vector Machines (RBF-SVM), was proposed to build the prediction model to identify the struggling students. Next, we employ an extra-tree classification method to rank the importance of these features so that teachers can integrate the importance ranking with the abstract task model to diagnose students' problems for scaffolding design. The overall goal of this study is to explore the feasibility of using predictive analytics in students' learning of engineering design.

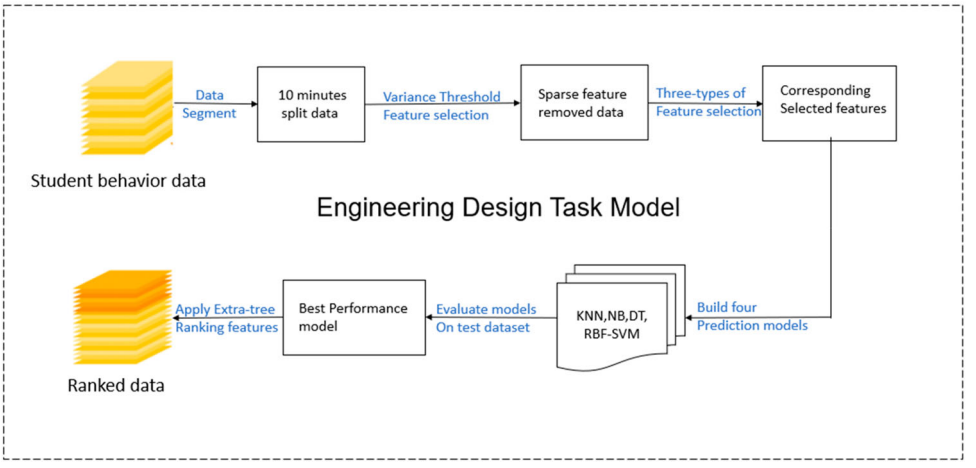


Figure 1. The methodology workflow.

Literature review

Engineering design

Engineering design has gained increased importance in education, as curriculum designers respond to studies establishing its relationship with science learning, twenty-first century skills, and academic motivation (Xie, Schimpf, Chao, Nourian, & Massicotte, 2018). However students have difficulty in mastering the necessary engineering design skills (Xie et al., 2014). One reason for this difficulty is that they lack frequent support and feedback from teachers to help them navigate the vast design space(Zheng et al., 2020). But in a classroom with 20–30 students working simultaneously on an open-ended design task, it is extremely difficult for teachers to spot the students who need support. Larger classes and those that take place online compound the problem.

Current research and practice have identified several ways to assess students' engineering designs and provide feedback. For instance, Atman, McDonnell, Campbell, Borgford-Parnell, and Turns (2015) employed verbal protocol analysis, asking students to think aloud while working on a design task. Transcripts of these thoughts allowed them to differentiate novice and expert student designers and provide more support to the novice designers. Gast, Lloyd, and Ledford (2018) requested students write a design report to log the specific steps they used in the design process for assessment. Purzer, Goldstein, Adams, Xie, and Nourian (2015) used concepts maps to assess students' understanding of engineering design to identify areas where students required support. Rehfeldt, Jung, Aguirre, Nichols, and Root (2016) implemented a peer-critique strategy for scoring and feedback in which he asked students to evaluate other students' design process. Timms and his colleagues (2018) asked students to record their design rationale using a graphical software program, Compendium. Although they used technology to organize the data and support the assessment process, the student designers themselves retained ultimate responsibility for reporting the design process. Thus, Timms et al. identified a method for self-assessment and reflection instead of providing direct teachers' support.

Past studies have varied in the engineering design frameworks they used to transform the collected data to assess students' engineering design. Gero's function-behavioral-structure design model (FBS), which provides an ontology behind various engineering design models, is a fundamental engineering design framework (Gero & Kannengiesser, 2014). Other frameworks, including sequential design, centered, concurrent, or dynamic engineering design models (Holmes & Yazdani, 1999; Yazdani, 1999) fit into the FBS ontology. FBS can serve as a base for many models that aim to assess students' learning of engineering design. Sequential engineering design model

indicates that the design follow series of certain steps (Holmes & Yazdani, 1999). It usually starts with exploration, and information gathering. Then, the design of the engineering produce would take place. After the design was complete, the process of verification and modification would be created. Most research studies that use these engineering design models assess students' understandings and skills of engineering design by analyzing their behaviors and design documents against the different aspects of the models. Ideally, teachers can then use the assessment results to provide personalized feedback to students.

A major limitation for these existing assessment methods is that they are mainly manual approaches, which requires time-consuming data collection and analysis procedures. It is difficult, if not impossible, to analyze the data from a class of students and provide timely feedback. Therefore, these methods are hard to scale. Worse, these assessment results are usually conducted after students have completed their engineering design tasks. Thus, they do not provide support for students during the design process.

LA and EDM

LA and EDM have been applied widely in the education field in recent years. These approaches exploit statistical, machine learning, and data mining algorithms to analyze various types of educational data (Papamitsiou & Economides, 2014). The main aim of both LA and EDM is to analyze educational big data to resolve educational issues and optimize learning and the learning environment (Snead & Freiberg, 2017). For instance, they can be used to model student behavior, predict student performance, increase self-reflection and awareness, improve assessment and feedback, and recommend resources (Papamitsiou & Economides, 2014; Xing & Goggins, 2015; Xing & Du, 2019). Studies have also used LA and EDM to visualize data, group students, develop concept maps, construct courseware, and schedule of classes (Khare, Lam, & Khare, 2018).

A few studies have explored the usage of LA and EDM in engineering design. Worsley and Blikstein (2013) combined LA with qualitative analysis to understand the open-ended engineering design tasks by developing a fine-grained representation of how experience relates to engineering practices. Dong, Hill, and Agogino (2004) employed computational methods known as latent semantic analysis to characterize students' design process. Vieira, Goldstein, Purzer, and Magana (2016) used learning analytics to discover student experimentation strategies in engineering design. Xie et al. (2018) applied time-series analytics to process low-level trace data, measuring students' engagement in engineering design, revealing gender differences, and distinguishing design cycles.

Studies of intelligent tutoring systems have usually constructed a task model to link the students' behavioral features in the learning environment with the abstract learning process for assessment (Arastoopour, Swiecki, Chesler, & Shaffer, 2015; Chen et al., 2017; Darabi, Arrington, & Sayilir, 2018; González-Brenes & Mostow, 2012). In this way, teachers and/or the system can provide specific feedback to students to improve their learning. The task models are usually modeled using probability-based algorithms such as the Bayesian network and the dynamic Bayesian network (Desmarais & Baker, 2012). These algorithms are white box models (Romero, Olmo, & Ventura, 2013) but need content experts to subjectively define initial parameters. Therefore, this task modeling method is a top-down approach and unable to support black-box machine learning algorithms. In addition, the fact that the white-box Bayesian approach is subjective undermines its generalizability. Previous Bayesian models (e.g. learning management system, online courses, and intelligent tutoring systems) are usually built upon 5–20 data features and the task model is much simpler than what need to be built in engineering design (Kloft, Stiehler, Zheng, & Pinkwart, 2014). The data sparsity and high-dimensionality in engineering design will result in a very complex Bayesian network which is difficult to train and interpret.

Methodology

Research context and data

Energy3D is an engineering tool created to provide students hands-on experience in designing and testing prototype projects harnessing renewable energy from middle school to the graduate level (Xie et al., 2018). With its easy to manage yet powerful design features and simulation capacity, it also appeals to some renewable energy industry practitioners who use it to perform entry-level science calculations. For example, Energy 3D makes it possible to explore the electricity production of solar panels in a particular positioning and arrangement on top of a house roof by building a virtual house model, setting its location, laying out the surroundings, arranging solar panels, and performing energy analyses. Figure 2 is a screenshot of the Energy3D software user interface with a solarized house and some analysis results represented by a time graph.

In this study, students were given a realistic engineering problem: to design an "energy-plus house" for the greater Boston area – an house that would conform to requirements such as height of wall, window/area ratio, and budget and that would have solar panels that produce more energy than it consumes on a yearly basis Students were to create three independent energy-plus house designs using three different styles: Cape Cod (Figure 3), Colonial, and Ranch. The three styles come with different aesthetics, construction cost, and energy efficiency, while successful solutions require students to possess and apply relevant energy-related knowledge in various realistic settings. The research design of asking students to complete three different, yet closely related tasks in a row provides the flexibility to study their performance on each individual task, as well as to observe how they transfer what they learn to other circumstances. During the project implementation, students were first introduced to Energy3D and got familiar with its essential functionalities for a whole session (about 45 min). Six class sessions were then allocated to them to complete the three designs, with two sessions for each design task. During the design process, students were to iterate the planning, building, evaluating, and optimizing stages to prepare a viable solution so that students can experience the cycles of engineering design to improve their engineering design practices. Study participants consisted of 111 high school students enrolled in a 9th grade physical science course. Three students' log files were incomplete and 108 students are remaining. All the digital log data from the students were collected at the end of the design task.

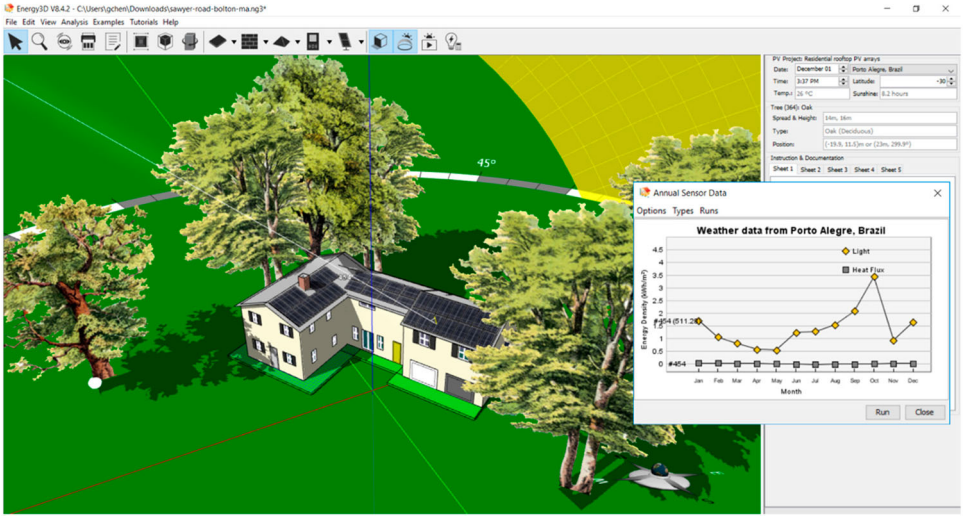


Figure 2. A screenshot of the user interface of Energy3D.

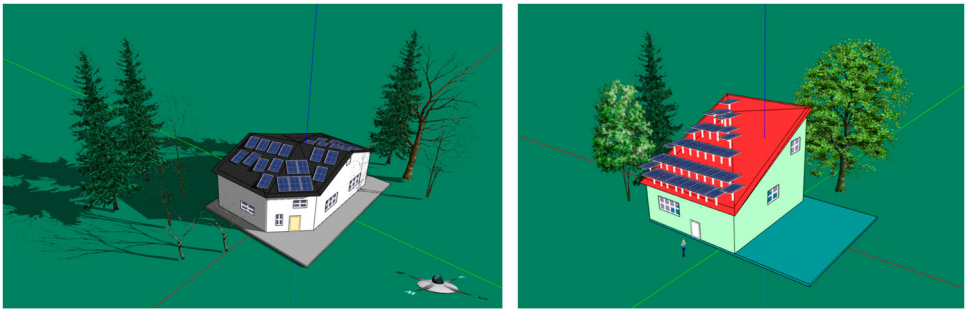


Figure 3. Student-generated Cape Cod style houses with solar panels on top of the roof.

Engineering design task model construction

In order to organize students' observable actions in ways that are informative to instructors in terms of design thinking, a task model (Segedy et al., 2014) was built and refined to provide a lens through which various design pathways could be evaluated. This model is broadly derived from the FBS model and the sequential engineering design model mentioned in the literature review section (Gero & Kannengiesser, 2014; Holmes & Yazdani, 1999; Yazdani, 1999). A particular action or a combination of actions may exist in different time segments with a purpose as intended by a student, which is closely related to the design process.

As displayed in Figure 4, students manage to solve the given problem by engaging in five distinct stages of the design process, which are governed by three broad cognitive classes. This engineering task model is broadly based on FBS model to discuss the relationship between function, behavior and structure. Then the design phase and the cognitive classes are then derived mainly from the sequential engineering design model. A student tackles the given problem by first engaging in acquiring relevant information such as date and time, sun path, annotations, etc. After figuring out useful information, a student enters the construction stage to generate an initial solution. The solution is then carried forward to the testing stage, within which a series of analyses are performed to produce results for solution evaluation. Additional information obtained through evaluation could guide the process of construction and revision cycle to refine a particular solution (Gero & Kannengiesser, 2014). The design process is iterative in its nature and sometimes a student needs not only to revise

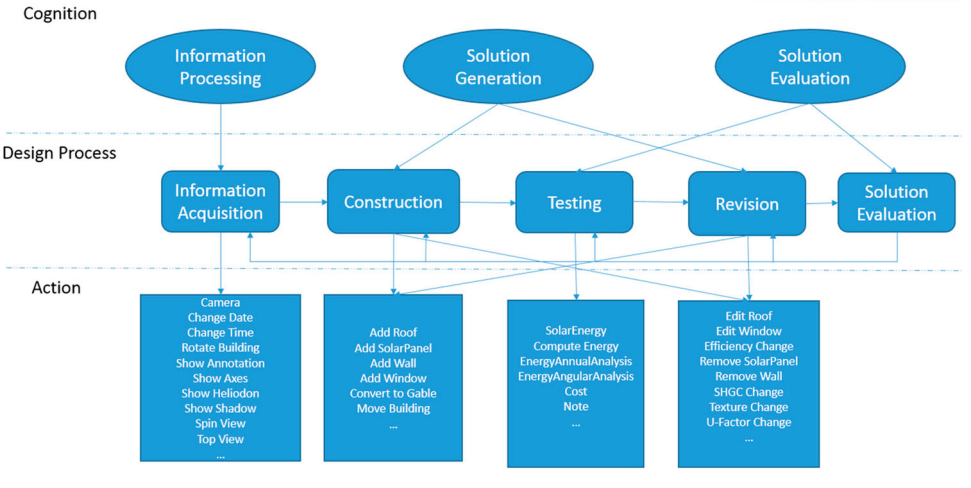


Figure 4. Engineering design task model.

an existing solution based on evaluation results but to acquire more relevant and accurate information to dispel misconceptions, confirm the validity of actions, and the facilitate decision-making process in order to renovate the solution.

Such a task model can direct a systematic interpretation and analysis of students' actions as logged by Energy3D to evaluate their individual design trajectory. For example, when a student is editing solar panels on a rooftop, this action could be interpreted as being related to revising a given design based on a previous round of info collection. In making such a revision students are guided by their intention to refine an existing solution. The abstraction of specific actions into design thinking processes could help teachers and researchers to understand how those actions contribute to success or failure in Energy3D exclusive problems as well as other engineering design tasks of the same type.

Data preprocess

This study focuses on the students' design of the Cape Cod style houses. In total, these students used 95 different features such as edit walls, show annotations, and edit solar panels. In order to build a model that can identify the optimum time point to predict student performance, we segmented each student's digital data traces into 10-minute parts. Then for each 10-minute segment, we computed the frequency of each action. The final performance of each student is represented by 1(succeed) or 0 (failure) to show whether the energy consumed by the student's Cape Cod house would be higher than zero. Summing all 95 features shows the average feature value is 1634.144 with a range of [1, 86,852], standard deviation of 9,142.262. Compared with previous prediction models, 95 features can be considered a very highly dimensional seta of data with which to build a prediction model, especially considering the small sample size. If we consider 10 segments for a student, then the feature space can reach 950 dimensions. Given the data range and its standard deviation, we can infer that the data is very sparse as well. Since students have different ways to design, they use quite different features. As a result, any given feature can be very sparse, with a lot of 0s for each student.

Table 1 shows the sample feature data set. These features such as "Edit.Wall" and "Edit.solarPanel" can have very different scales and contain large outliers. Having features in different scales can make it very difficult to visualize. More importantly, they can degrade the performance of the prediction model since the outliers and different scale may dominate prediction in an unknown direction (Jain, Bhatele, Robson, Gamblin, & Kale, 2013). Therefore, standard scaler method was used to transform all the features in this part. Standardization of a data set is a common requirement for many machine learning algorithms. It centers and scales the data set on each feature independently by removing the mean and scaling to the unit variance (Schoenfeld, Giraud-Carrier, Poggemann, Christensen, & Seppi, 2018).

Two-stage feature selection

Sparsity and high dimensionality in engineering design can affect the performance of a prediction model significantly (Szegedy et al., 2015) and result in task modeling results that are complex for

Table 1. Sample feature data set.

Student	Segment#	Add.Wall	Edit.Wall	Edit.SolarPanel	…	Feature 95	Performance
S1	0	7	0	0	…	…	1
S1	1	1	15	0	…	…	1
S1	2	0	0	0	…	…	1
S1	3	0	0	22	…	…	1
…							
S2	0	6	7	0	…	…	0
S2	1	0	1	13	…	…	0
…							

teachers to interpret. Therefore, we propose a two-stage feature selection method to reduce the feature sparsity and the high dimensionality.

In the first stage, the variance threshold feature selection method was used to reduce the sparsity. As a classic sparsity reduction method, the variance threshold method belongs to the family of filtering feature selection approaches. A filtering method is used to evaluate the values in the feature space and select the best features before feeding any data to the model shown as in Figure 5(a). Entropy and information gain are commonly used to measure the importance of certain features in filtering. That is, features with the variances that are lower than the assigned threshold will be removed (Henni, Mezghani, & Gouin-Vallerand, 2018).

In the second stage, three different feature selection methods were applied to further reduce the high dimensionality, including univariate feature selection, recursive feature elimination, and principle component analysis (PCA). These feature selection methods belong to the wrapping family of feature selection approaches (|Méndez-Cruz et al., 2017). It is usually accomplished by testing a number of reduced feature-spaces on the model and checking which feature performs best, as shown in Figure 5(b). This method usually combines with some prediction or classification models for feature selection. Therefore, it is well suited for dimension reduction and improves the prediction performance.

Specifically, univariate feature selection evaluates the strength of each feature in the data set individually to select those features that have significant relationships with the labels of the data set (Nielsen et al., 2018). Chi-square function was used to identify the features. Recursive feature elimination is a feature selection method that recursively selects features from the data set to fit the model until all the weak features are removed (Chandrashekar & Sahin, 2014). PCA is a widely used dimension-reduction tool that can be used to compress a set of correlated data into a smaller set that still contains the most important information in the original data (Abdi & Williams, 2010). This method selects those features that have a strong relationship with the principal component in the data set.

RBF-SVMs

As probability-based algorithms require subjective definition of the parameters, in this study, we proposed to use a bottom-up approach, supervised machine learning, to represent the task model. RBF-SVM aims to identify the underperforming students first. Support Vector Machines (SVMs) are a type of popular supervised machine learning algorithm frequently used to build prediction models. They work well for data sets with complex data features (LeCun, Bengio, & Hinton, 2015). For the binary classification task, the algorithm will find an optimal hyperplane with the largest margin to separate

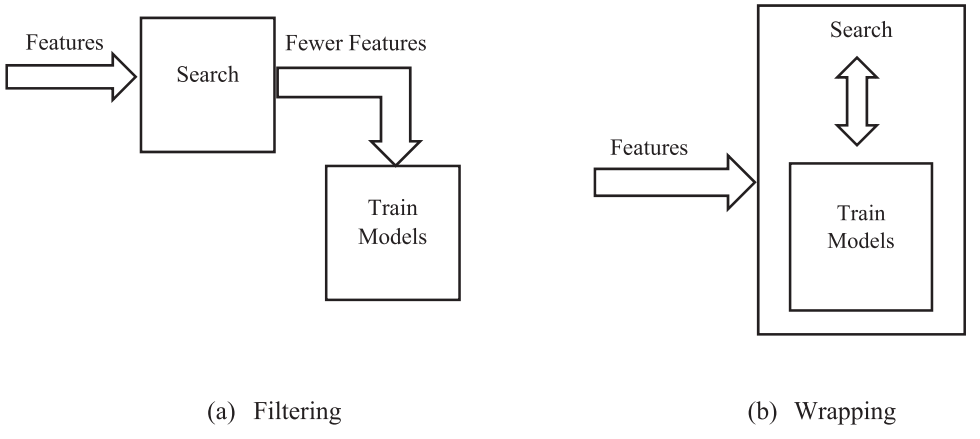


Figure 5. Comparison of different selection methods.

the two classes. Let $X = \{x_1, x_2, \ldots x_n\}$ with n data samples having corresponding labels $Y = \{\pm 1\}$, and the hyperplane used to separate the two classes is:

$$f(x) = \text{sign}(w \cdot x + b)$$

If the data set is separable, the data will be correctly separated if $y_i(w \cdot x_i + b) \geq 0\ \forall i$. In an actual data set, the data are not always linearly separable, so we cannot find an appropriate hyperplane to separate the data set in the original feature space. One common way is to transform the original data set into a higher feature space based on a commonly used method (kernel function). The kernel function can project the initial data in a feature space to a higher dimensional space $\emptyset{:}K^n \rightarrow H$. In this new space H, we hope to find a hyperplane to classify the dataset and then project back to the original feature space. The RBF-SVM has an excellent performance with great efficiency with the high dimensional data, and this allows us to easily find a hyperplane in the high feature space to separate the data. The RBF form adopted in this paper is shown as the following equation:

$$K(x_1, x_i) = \exp^{-\frac{|x - x_i|^2}{2*\delta^2}}$$

In order to demonstrate the good performance of the RBF-SVM classification method in our context, we also implemented three commonly used baseline models including K-Nearest Neighbor (KNN), Naïve Bayes, and Decision Tree. Details of these algorithms appear in Kotsiantis, Zaharakis, and Pintelas (2007). The data set was divided into a training set (70%) used to build the models and a testing set (30%) used to evaluate the performances of the models. Metrics like precision, recall, and F-measure were used to measure the performance of different models. In addition, the receiver operating characteristic (ROC) and area under the receiver operating characteristics curve (AUC) were also employed to compare the performance of these models.

Extra-tree classification for feature importance ranking to facilitate feedback design

After RBF-SVM identified the struggling students, the next step is to determine why students are struggling and design personalized help to these students. In this paper, we employ extra-tree classification to rank the feature importance (Suarez-Tangil et al., 2017). In this way, the teacher can integrate the abstract task model with the feature importance to understand why the student is struggling and then provide personalized feedback. In this algorithm, a cluster of trees are built iteratively on the dataset until each tree represents a sub-dataset. The nodes in the trees represent features of the dataset and the leaves represent the samples belong to a specific class. The trees are generated by splitting the nodes, and the features that are most related to the target will be split first based on the information gain theory. In this way, after the trees are generated, we order the nodes from the root to the leaves (exclude the leaf nodes) to get the feature sequence that ordered by the significances to target. After the features were ordered, feature importance is calculated by weighting the proportion of the samples that reaches a node in the whole data set with the purity of the node. In this paper, we ran the extra-tree classification approach multiple times and then average the feature importance score calculated by each iteration as the final score of feature importance.

Results

Scaling and feature selection results

Because many features were not in the same scale and there were very large outliers, standard scaler method was used to scale the original features. Figure 6 compares the data distribution of three features (“Change.Solar,” “Edit.Sensor,” and “Edit.Window”) before and after the scaling. The results show that the scaling features such as “Change.Solar” and “Edit.Sensor” are in a smaller range of

[0,3] and [0, 10] than "Edit.Window," which is in [0,70]. After the scaling, these features all fall into a similar range, about [0,15], as shown in Figure 6.

This paper further employed a two-stage feature selection method. The first stage was to implement the variance-threshold feature selection method to reduce its sparsity. Setting the threshold at 0.8 removed the lower variance features. In other words, features where most observations are 0s were not considered. The heat-map of Figure 7 shows the correlation of the remaining 58 features. The X-axis and Y-axis of the figure are features that are both in the same order. The color in the figure corresponds to the correlation degree between two features. The brighter the color, the higher the correlation between the two features. The diagonal of the figure shows the highest brightness, which means the same features have the highest correlation. This result indicated that even after removing the sparse features, many features had high collinearity.

In the second stage, three feature selection methods – univariate feature selection, recursive feature selection, and PCA methods – were applied to further reduce the dimension in order to keep features with low correlation distinct. In order to demonstrate the effectiveness of the two-stage feature selection, these three methods were first directly applied to all 95 features and then applied to the 58 features after removing the sparse features. Figure 8 shows the results. As shown, 20 features were selected by these methods. By comparing the feature selection results before and after reducing the sparse features, we found that there are more squares with a higher brightness in Figure 8(1) than Figure 8(2), which means that the features at those corresponding positions still have a higher correlation without the variance-threshold feature selection method.

Prediction performance

To further demonstrate the effectiveness of the two-stage feature selection and the robust performance of RBF-SVM, we conducted a prediction experiment on the selected features with and without variance threshold method (reducing sparsity). Various performance measures, including precision,

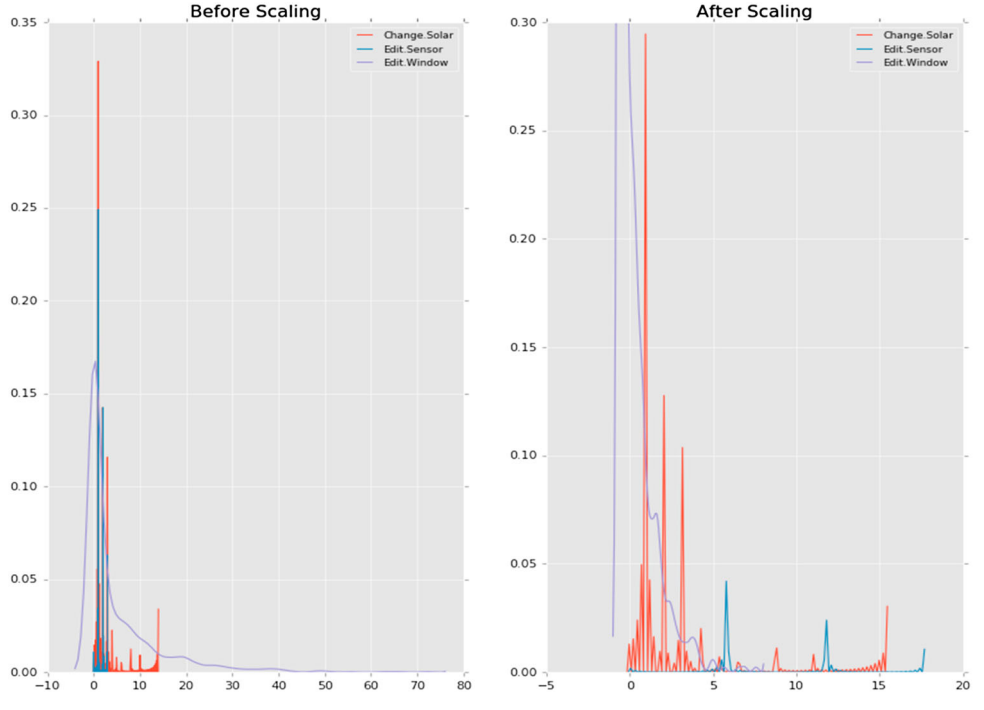


Figure 6. Comparison of the distribution of three different features before and after scaling.

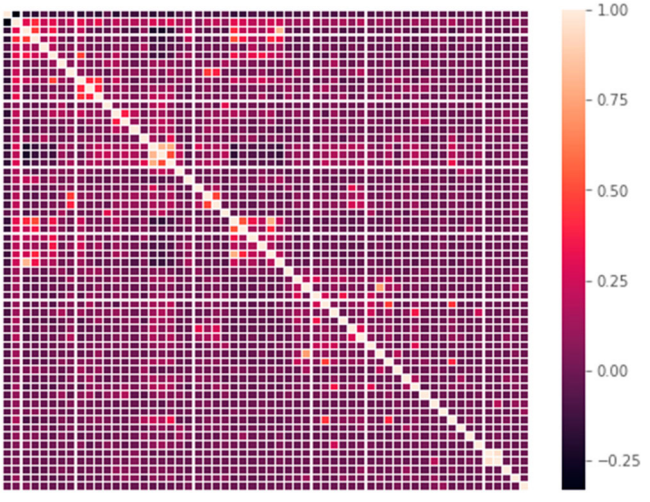


Figure 7. Heat-map of the correlations between the features after the first stage feature selection.

recall, F-measure as shown in Tables 2 and 3, ROC, and AUC are also plotted to demonstrate the prediction performance as shown in Figure 9.

Table 2 and Figure 9(1) showed the prediction modeling results of the proposed algorithm in comparison with the benchmark algorithms on the feature selection results without using the variance threshold method. The results reflected that RBF-SVM consistently performed better than the benchmark algorithms over the three feature selection methods. Specifically, RBF-SVM has the best

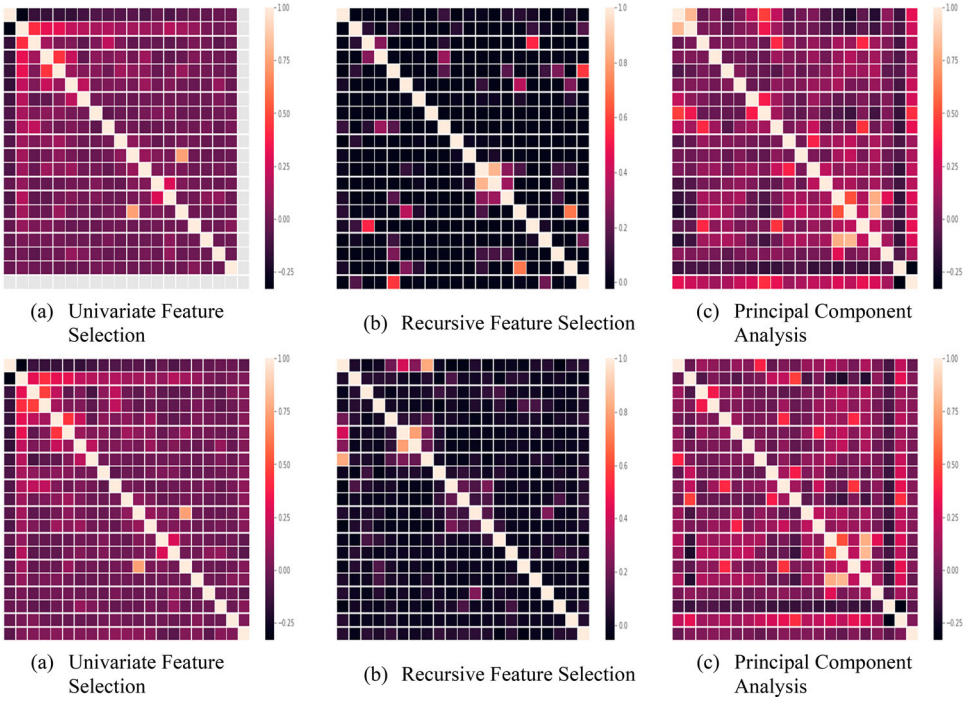


Figure 8. Heat-map of the correlation between the features after the second stage feature selection.

Table 2. The prediction performance on the three feature sets without reducing sparse features.

Data Set	Classifiers	Precision	Recall	*F*-measure
Univariate Feature Selection	KNN	0.645	0.700	0.655
	Naïve Bayes	0.534	0.591	0.506
	Decision Tree	0.555	0.801	0.518
	RBF-SVM	0.729	0.898	0.759
Recursive Feature Selection	KNN	0.623	0.670	0.631
	Naïve Bayes	0.527	0.541	0.391
	Decision Tree	0.555	0.717	0.524
	RBF-SVM	0.689	0.852	0.707
PCA	KNN	0.637	0.668	0.645
	Naïve Bayes	0.546	0.561	0.543
	Decision Tree	0.561	0.671	0.542
	RBF-SVM	0.783	0.915	0.817

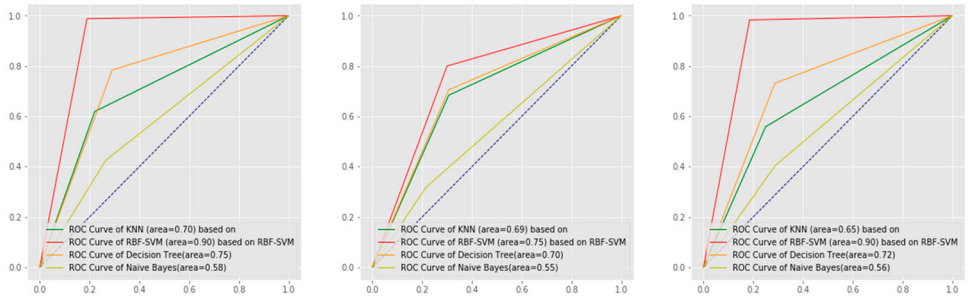
performance for the PCA feature selection results (Precision = 0.75, Recall = 0.899, *F*-measure = 0.781). The range for RBF-SVM is precision [0.53, 0.75], recall [0.75, 0.899], and F-measure [0.474, 0.781]. For the benchmark algorithms, KNN performed best in the univariate feature selection method (Precision = 0.665, Recall = 0.699, *F*-measure = 0.675). The range of the benchmark algorithms is precision [0.514, 0.665], recall [0.551, 0.699], and F-measure [0.289, 0.675]. RBF-SVM performed about 10% better for all the benchmark algorithms.

Table 3 and Figure 9(2) showed the prediction modeling results of the proposed algorithm in comparison with the benchmark algorithms on the feature selection results using the variance threshold method. The results indicated that RBF-SVM still performed better than the baseline algorithms over all the feature selection methods. To illustrate, RBF-SVM has the best performance for the PCA feature selection results (Precision = 0.783, Recall = 0.915, *F*-measure = 0.817). The range for RBF-SVM is precision [0.689, 0.783], recall [0.852, 0.915], and F-measure [0.707, 0.817]. For the benchmark algorithms, KNN performed best in the univariate feature selection method (Precision = 0.645, Recall = 0.700, *F*-measure = 0.655). The range of the benchmark algorithms is precision [0.527, 0.645], recall [0.541, 0.699], and F-measure [0.391, 0.675]. Similarly, in this full two-stage feature selection condition, RBF-SVM still performed about 10% better than the baseline algorithms.

Comparing the prediction results between full two-stage feature selection results and the other condition reveals that the two-stage feature selection results performed about 5% better than the other condition for most of the algorithms. In sum, two-stage feature selection has an advantage over other feature selection methods, but for each individual stage of feature selection, different algorithms may have different performance over specific feature selection results. RBF-SVM has better and more robust performance than all the benchmark algorithms. The best performance of RBF-SVM is Precision = 0.783, Recall = 0.915, and *F*-measure = 0.817, which is considered good to excellent according to the LA and EDM literature (see for example, Herrero, 2014).

Table 3. The prediction performance on three feature sets with reducing sparse features.

Data Set	Classifiers	Precision	Recall	*F*-measure
Univariate Feature Selection	KNN	0.665	0.699	0.675
	Naïve Bayes	0.578	0.580	0.579
	Decision Tree	0.550	0.747	0.512
	RBF-SVM	0.743	0.899	0.774
Recursive Feature Selection	KNN	0.515	0.689	0.443
	Naïve Bayes	0.510	0.551	0.289
	Decision Tree	0.514	0.699	0.441
	RBF-SVM	0.530	0.750	0.474
PCA	KNN	0.617	0.654	0.623
	Naïve Bayes	0.547	0.558	0.547
	Decision Tree	0.558	0.722	0.530
	RBF-SVM	0.750	0.899	0.781

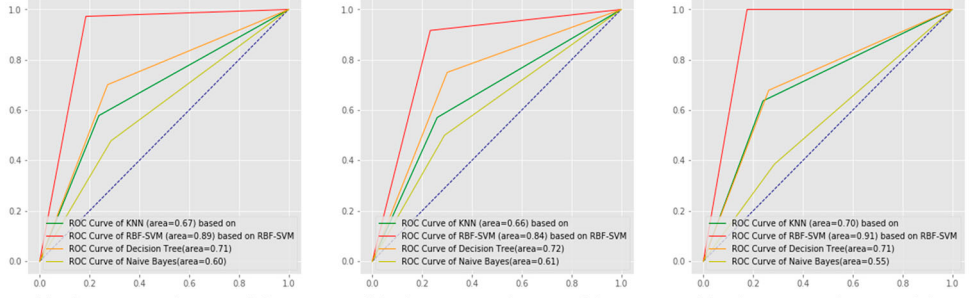


(a) The ROC and AUC of the classifiers based on univariate feature selection

(b) The ROC and AUC of the predictors based on recursive feature selection

(c) The ROC and AUC of the predictors based on PCA

(1) The prediction performance on the three feature sets without reducing sparse features



(a) The ROC and AUC of the classifiers based on univariate feature selection

(b) The ROC and AUC of the predictors based on recursive feature selection

(c) The ROC and AUC of the predictors based on PCA

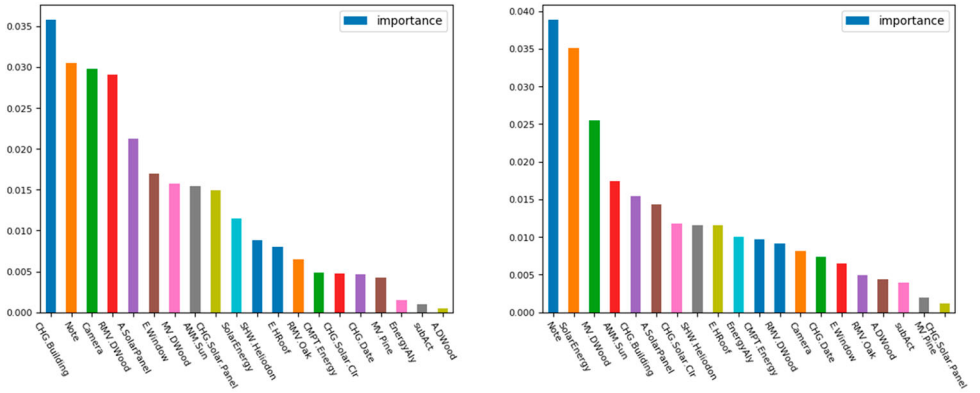
(2) The prediction performance on the three feature sets with reducing sparse features

Figure 9. The comparison of ROC of the algorithms on three feature selection sets.

Extra tree classification for feature importance ranking to facilitate feedback design

After RBF-SVM identified the struggling students, extra tree classification was further applied to rank the significance of the various features to support teachers to design scaffolding and support strategies. Compared with the 95 original features, the two-stage feature selection method has reduced the feature set to 20 features only. Therefore, the extra-tree classification was only applied to the 20 selected set. In this way, the results are much easier to understand than the traditional Bayesian network with possible 95 nodes. The results were presented in Figure 10 for the feature importance ranking. Overall, the feature importance ranking results can help teachers understand and estimate which specific phases of engineering design the students are currently working on and the main factors that contribute to students' success.

We chose two scenarios to demonstrate the usefulness of the feature ranking results. For instance, at minute 20, we identify that student A is struggling in the engineering design task using the RBF-SVM. The feature ranking results showed that change.buiding, camera, and add edit.window were generally rated much higher than the rest. The teacher can then estimate the student is in the information acquisition and construction phase of engineering design according to the engineering design task model. Also, teachers or an intelligent agent can approach the students by providing scaffolding and help related with building construction such as automatically receiving a pop-up hint, "Do you need help on the building construction?" or even automatically showing a list of useful resources to prompt student to gather information to guide their building construction. It can go even further to sequence the resources based on the ranking of feature importance. At minute 40, RBF-SVM identified that student B is having difficulty undertaking the design challenge.



(a) Feature Importance Ranking at 20 minutes (b) Feature Importance Ranking at 40 minutes

Figure 10. Feature importance ranking scenarios.

Given the feature ranking of SolarEnergy, change.solar and energyanalysis are higher but other actions are ranked much lower; the teacher can estimate that students are engaged in the revision and testing phase. Similarly, teachers or an artificial intelligent agent can provide relative support for student B to overcome his or her difficulties.

Discussion

Despite the importance of engineering design in education, it is still very challenging for students to learn and master the related concepts and knowledge (Atman, Kilgore, & McKenna, 2008). Because of the open-ended nature and complexity of learning engineering design, lessons based on it make it difficult for both researchers and practitioners to assess students and provide timely support and scaffolding to them. Traditional methods, such as analyzing verbal protocol, designing documents, and using concepts maps (Kilsdonk, Peute, Riezebos, Kremer, & Jaspers, 2016) to assess the engineering design process and students' performance may not provide a reliable measure or allow for early intervention.

The ability of engineering design platforms to log low level student trace data opens up opportunities to use LA and EDM to build prediction models to automatically identify struggling students. Compared with traditional manual methods, such predictive analytics have two advantages. First, the whole process is automatic and can be easily scaled up to large classes and online contexts. Second, given the early detection requirement for intervention design, predictive modeling can identify at-risk students before they fail the task. By contrast, traditional methods usually assess students after they finish the engineering design work and thus cannot support early intervention. While there are some initial studies of LA and EDM in engineering design (Pedullà et al., 2016), this is the first study that aims to explore how to build prediction models to provide timely support for students.

From a methodological perspective, it is challenging to construct a workflow for predictive analytics to support engineering design. This is mainly because of the sparsity and high dimensionality in the data streams generated by engineering design. This study proposed a practical workflow and experimented with various approaches for prediction model construction. To address the data sparsity and high dimensionality, a two-stage feature selection method was proposed. In our experiment, this two-stage feature selection effectively addressed the data issues and the prediction results outperformed the models built using other feature selection methods. Moreover, in lieu of the traditionally used algorithms, this study proposed to use RBF-SVM, which fits the data very well and has better performance for the benchmark algorithms such as KNN, Naïve Bayes, and Decision Tree. The overall

workflow generated good to excellent performance compared with other prediction models in the LA and EDM literature (Long, Smith, Dass, Dillon, & Hill, 2016).

Traditional task modeling for learning analytics usually uses a top-down probabilistic framework such as a Bayesian network (Desmarais & Baker, 2012; González-Brenes & Mostow, 2012). This approach requires significantly subjective judgment from the content experts and is difficult to handle for large feature sets. By comparison, the two-stage feature selection proposed here can significantly reduce the modeling complexity in terms of both computing cost and the teachers' interpretation. The RBF-SVM is a data-driven, bottom-up approach to assessment with a more objective perspective. The following extra-tree classification method to rank feature importance is able to support concrete intervention and scaffolding strategies design. This study thus provides a new approach for task modeling, which is an essential development for dynamic assessment.

There are several limitations associated with this study. First, this study is built only for one engineering design task in a specific engineering design platform. It should be used with caution if using the prediction modeling methods and results in other tasks and contexts. Second, this study is a very exploratory step for predictive analytics in engineering design. The results are mainly derived from the limited collected data for this one experiment and usefulness in a live classroom is not yet proven.

Conclusion

This study takes an initial step toward a method of early and accurate identification of struggling students in learning engineering design. Specifically, this work first developed a task model to associate students' behavioral features with the abstract design process and then proposed a two-stage feature selection method to address the data sparsity and high dimensionality issues in engineering design. Next, RBF-SVM and extra-tree classification method was employed for the engineering design task modeling. Results showed that the combination of two stage feature selection and RBF-SVM can achieve very good prediction performance in identifying struggling students and have an advantage over the baseline models. The extra-tree classification results to rank the feature importance can further provide a concrete and quantified base for teachers to design scaffolding and support strategies to help the students.

This work also builds a foundation for further exploration. First, there is still a lot of room to continue optimizing the prediction performance for engineering design. Future studies can experiment more with different feature engineering methods and algorithms to improve the prediction accuracy. Second, the prediction results of the current model only reveal success and failure as summative measures. Future research should incorporate more measures to better represent engineering design so that the prediction models can consider more performance measures of engineering design. Third, the current estimation of engineering design phase depends on the feature ranking results along with teachers' engineering design knowledge. Future work can explore more quantitative and objective ways for such estimation. Fourth, future research should implement the prediction model in an actual engineering design class and examine its effectiveness in supporting students' learning.

Acknowledgements

This work is supported by the National Science Foundation (NSF) of the United States under grant numbers 1512868, 1348530 and 1503196. Any opinions, findings, and conclusions or recommendations expressed in this paper, however, are those of the authors and do not necessarily reflect the views of the NSF. The authors are indebted to Joyce Massicotte, Elena Sereiviene, Jie Chao, Xudong Huang, and Corey Schimpf for assistance and suggestions.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work is supported by the Directorate for Education and Human Resources in National Science Foundation (NSF) of the United States under [grant numbers 1512868, 1348530 and 1503196].

Notes on contributors

Wanli Xing is an Assistant Professor of Educational Technology at University of Florida. His research interests are artificial intelligence, learning analytics, STEM education and online learning.

Bo Pei is a PhD student in Educational Technology at University of Florida. He earned his bachelor and master's degree in computer science. His research interests include educational data mining, image processing, and machine learning.

Shan Li is a doctoral student in the Department of Educational and Counselling Psychology at McGill University. He is currently a research assistant in the ATLAS (Advanced Technologies for Learning in Authentic Settings) Lab. His research focuses on the cognitive and metacognitive aspects of self-regulated learning within technology-rich learning environments, taking advantage of eye-tracking and data mining techniques to develop a deep understanding of students' self-regulatory learning processes.

Guanhua Chen is a postdoctoral research associate at The Concord Consortium. He earned his PhD from the University of Miami with a background and expertise in science education, educational technology and data mining.

Charles Xie is a senior scientist at the Concord Consortium. His research includes computational science, learning science, machine learning, artificial intelligence, computer-aided design, mixed reality, Internet of Things, molecular simulation, solar energy engineering, and infrared imaging. He develops the Energy3D software that supports this research.

ORCID

Wanli Xing http://orcid.org/0000-0002-1446-889X

References

Abdi, H., & Williams, L. J. (2010). Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, *2*(4), 433–459.

Arastoopour, G., Swiecki, Z., Chesler, N. C., & Shaffer, D. W. (2015). *Epistemic network analysis as a tool for engineering design assessment*. Seattle, WA: American Society for Engineering Education.

Atman, C. J., Adams, R. S., Cardella, M. E., Turns, J., Mosborg, S., & Saleem, J. (2007). Engineering design processes: A comparison of students and expert practitioners. *Journal of Engineering Education*, *96*(4), 359–379.

Atman, C. J., Kilgore, D., & McKenna, A. (2008). Characterizing design learning: A mixed - methods study of engineering designers' use of language. *Journal of Engineering Education*, *97*(3), 309–326.

Atman, C. J., McDonnell, J., Campbell, R., Borgford-Parnell, J., & Turns, J. (2015, June). *Using design process timelines to teach design: Implementing research results*. American Society of Engineering Education Conference.

Author. (2016).

Baker, R. S. J. D., & Gowda, S. M. (2010). *An analysis of the differences in the frequency of students' disengagement in urban, rural, and suburban high schools*. Proceedings of the 3rd international conference on educational data mining (pp. 11–20).

Chandrashekar, G., & Sahin, F. (2014). A survey on feature selection methods. *Computers & Electrical Engineering*, *40*(1), 16–28.

Chen, Y., Liu, Q., Huang, Z., Wu, L., Chen, E., Wu, R., … Hu, G. (2017, November). *Tracking knowledge proficiency of students with educational priors*. Proceedings of the 2017 ACM on Conference on Information and Knowledge Management (pp. 989–998), ACM.

Darabi, A., Arrington, T. L., & Sayilir, E. (2018). Learning from failure: A meta-analysis of the empirical studies. *Educational Technology Research and Development*, *66*(5), 1101–1118.

Desmarais, M. C., & Baker, R. S. (2012). A review of recent advances in learner and skill modeling in intelligent learning environments. *User Modeling and User-Adapted Interaction*, *22*(1–2), 9–38.

Dong, A., Hill, A. W., & Agogino, A. M. (2004). A document analysis method for characterizing design team performance. *Journal of Mechanical Design*, *126*(3), 378–385.

Dutt, A., Ismail, M. A., & Herawan, T. (2017). A systematic review on educational data mining. *IEEE Access*, *5*, 15991–16005.

Dym, C. L., Agogino, A. M., Eris, O., Frey, D. D., & Leifer, L. J. (2005). Engineering design thinking, teaching, and learning. *Journal of Engineering Education*, *94*(1), 103–120.

Gašević, D., Dawson, S., & Siemens, G. (2015). Let's not forget: Learning analytics are about learning. *TechTrends*, *59*(1), 64–71.

Gast, D. L., Lloyd, B. P., & Ledford, J. R. (2018). Multiple baseline and multiple probe designs. In *Single case research methodology* (pp. 239–281). New York: Routledge.

Gero, J. S., & Kannengiesser, U. (2014). The function-behaviour-structure ontology of design. In *An anthology of theories and models of design* (pp. 263–283). London: Springer.

González-Brenes, J. P., & Mostow, J. (2012). Dynamic cognitive tracing: Towards unified discovery of student and cognitive models. In *Proceedings of the 5th International Conference on Educational Data Mining* (pp. 49–56). Chania, Greece: International Educational Data Mining Society.

Henni, K., Mezghani, N., & Gouin-Vallerand, C. (2018). Unsupervised graph-based feature selection via subspace and pagerank centrality. *Expert Systems with Applications*, *114*, 46–53.

Herrero, G. (2014). *Towards supervised music structure annotation: A case-based fusion approach*.

Holmes, C., & Yazdani, B. (1999). *Internal drivers for concurrent engineering industrial case study*. 5th International Conference on Concurrent Enterprising (ICE'99) (pp. 455–464).

Jain, N., Bhatele, A., Robson, M. P., Gamblin, T., & Kale, L. V. (2013, November). *Predicting application performance using supervised learning on communication features*. Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis (p. 95), ACM.

Khare, K., Lam, H., & Khare, A. (2018). Educational data mining (EDM): Researching impact on online business education. In *On the line* (pp. 37–53). Cham: Springer.

Kilsdonk, E., Peute, L. W., Riezebos, R. J., Kremer, L. C., & Jaspers, M. W. (2016). Uncovering healthcare practitioners' information processing using the think-aloud method: From paper-based guideline to clinical decision support system. *International Journal of Medical Informatics*, *86*, 10–19.

Kloft, M., Stiehler, F., Zheng, Z., & Pinkwart, N. (2014). *Predicting MOOC dropout over weeks using machine learning methods*. Proceedings of the EMNLP 2014 Workshop on Analysis of Large Scale Social Interaction in MOOCs (pp. 60–65).

Kotsiantis, S. B., Zaharakis, I., & Pintelas, P. (2007). Supervised machine learning: A review of classification techniques. *Emerging Artificial Intelligence Applications in Computer Engineering*, *160*, 3–24.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, *521*(7553), 436–444.

Leu, D. J., Kinzer, C. K., Coiro, J., Castek, J., & Henry, L. A. (2017). New literacies: A dual-level theory of the changing nature of literacy, instruction, and assessment. *Journal of Education*, *197*(2), 1–18.

Long, R., Smith, M., Dass, S., Dillon, C., & Hill, K. (2016). *Data analytics: Techniques and applications to transform army learning*. ModSim World Conference, NDIA.

Méndez-Cruz, C. F., Gama-Castro, S., Mejía-Almonte, C., Castillo-Villalba, M. P., Muñiz-Rascado, L. J., & Collado-Vides, J. (2017). *First steps in automatic summarization of transcription factor properties for RegulonDB: Classification of sentences about structural domains and regulated processes*. Database.

National Assessment Governing Board. (2013). *Technology and engineering literacy framework for the 2014 national assessment of educational progress*. Washington, DC: ERIC Clearinghouse.

Nielsen, A. N., Greene, D. J., Gratton, C., Dosenbach, N. U., Petersen, S. E., & Schlaggar, B. L. (2018). Evaluating the prediction ofbrain maturity from functional connectivity after motion artifact denoising. *Cerebral Cortex*, *29*(6), 2455–2469.

Papamitsiou, Z., & Economides, A. A. (2014). Learning analytics and educational data mining in practice: A systematic literature review of empirical evidence. *Journal of Educational Technology & Society*, *17*(4), 49–64.

Pedullà, E., Savio, F. L., Boninelli, S., Plotino, G., Grande, N. M., La Rosa, G., & Rapisarda, E. (2016). Torsional and cyclic fatigue resistance of a new nickel-titanium instrument manufactured by electrical discharge machining. *Journal of Endodontics*, *42*(1), 156–159.

Peña-Ayala, A. (2014). Educational data mining: A survey and a data mining-based analysis of recent works. *Expert Systems with Applications*, *41*(4), 1432–1462.

Purzer, Ş, Goldstein, M. H., Adams, R. S., Xie, C., & Nourian, S. (2015). An exploratory study of informed engineering design behaviors associated with scientific explanations. *International Journal of STEM Education*, *2*(1), 9.

Rehfeldt, R. A., Jung, H. L., Aguirre, A., Nichols, J. L., & Root, W. B. (2016). Beginning the dialogue on the e-transformation: Behavior analysis' first massive open online course (MOOC). *Behavior Analysis in Practice*, *9*(1), 3–13.

Romero, C., Olmo, J. L., & Ventura, S. (2013, July). *A meta-learning approach for recommending a subset of white-box classification algorithms for Moodle datasets*. Educational Data Mining.

Schoenfeld, B., Giraud-Carrier, C., Poggemann, M., Christensen, J., & Seppi, K. (2018). Preprocessor selection for machine learning pipelines. arXiv preprint arXiv:1810.09942.

Schwarz, C. V., Reiser, B. J., Davis, E. A., Kenyon, L., Achér, A., Fortus, D., … Krajcik, J. (2009). Developing a learning progression for scientific modeling: Making scientific modeling accessible and meaningful for learners. *Journal of Research in Science Teaching*, *46*(6), 632–654.

Segedy, J. R., Biswas, G., & Sulcer, B. (2014). A model-based behavior analysis approach for open-ended environments. *Journal of Educational Technology & Society*, *17*(1), 272–282.

Snead, L. O., & Freiberg, H. J. (2017). Rethinking student teacher feedback: Using a self-assessment resource with student teachers. *Journal of Teacher Education*, *70*(2), 155–168.

Suarez-Tangil, G., Dash, S. K., Ahmadi, M., Kinder, J., Giacinto, G., & Cavallaro, L. (2017, March). *Droidsieve: Fast and accurate classification of obfuscated android malware*. Proceedings of the Seventh ACM on Conference on Data and Application Security and Privacy (pp. 309–320), ACM.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., … Rabinovich, A. (2015). *Going deeper with convolutions*. Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1–9).

Timms, M. J., Moyle, K., Weldon, P. R., & Mitchell, P. (2018). Challenges in STEM learning in Australian schools. *Literature and Policy Review*.

Vieira, C., Goldstein, M. H., Purzer, Ş, & Magana, A. J. (2016). Using learning analytics to characterize student experimentation strategies in the context of engineering design. *Journal of Learning Analytics*, *3*(3), 291–317.

Worsley, M., & Blikstein, P. (2013, April). *Towards the development of multimodal action based assessment*. Proceedings of the third international conference on learning analytics and knowledge (pp. 94–101), ACM.

Wu, X., Zhu, X., Wu, G. Q., & Ding, W. (2014). Data mining with big data. *IEEE Transactions on Knowledge and Data Engineering*, *26*(1), 97–107.

Xie, C., Schimpf, C., Chao, J., Nourian, S., & Massicotte, J. (2018). Learning and teaching engineering design through modeling and simulation on a CAD platform. *Computer Applications in Engineering Education*, *26*(4), 824–840.

Xie, C., Zhang, Z., Nourian, S., Pallant, A., & Hazzard, E. (2014). A time series analysis method for assessing engineering design processes using a CAD tool. *International Journal of Engineering Education*, *30*(1), 218–230.

Xing, W., Chen, X., Stein, J., & Marcinkowski, M. (2016). Temporal predication of dropouts in MOOCs: Reaching the low hanging fruit through stacking generalization. *Computers in Human Behavior*, *58*, 119–129.

Xing, W., & Du, D. (2019). Dropout prediction in MOOCs: Using deep learning for personalized intervention. *Journal of Educational Computing Research*, *57*(3), 547–570.

Xing, W., & Goggins, S. (2015, March). Learning analytics in outer space: a Hidden Naïve Bayes model for automatic student off-task behavior detection. In *Proceedings of the Fifth International Conference on Learning Analytics And Knowledge* (pp. 176–183). ACM..

Xing, W., Guo, R., Petakovic, E., & Goggins, S. (2015). Participation-based student final performance prediction model through interpretable Genetic Programming: Integrating learning analytics, educational data mining and theory. *Computers in Human Behavior*, *47*, 168–181.

Yazdani, B. (1999). Four models of design definition: Sequential, design centered, concurrent and dynamic. *Journal of Engineering Design*, *10*(1), 25–37.

Zheng, J., Xing, W., Zhu, G., Chen, G., Zhao, H., & Xie, C. (2020). Profiling self-regulation behaviors in STEM learning of engineering design. *Computers & Education*, *143*, 103669. doi:10.1016/j.compedu.2019.103669.

Zhou, L., Hu, Y., & Xu, J. (2018). Understanding the lack of student engagement in Chinese library science undergraduate education. *Information Development*, *34*(2), 148–161.