An Artificial Intelligence Approach to Identifying Skill Relationship

Tak-Lam WONG*, Yuen Tak YUb, Chung Keung POONc, Haoran XIEd, Fu Lee WANGc & Chung Man TANGb

aDepartment of Computing Studies & Information Systems, Douglas College, Canada
bDepartment of Computer Science, City University of Hong Kong, Hong Kong
cSchool of Computing and Information Sciences, Caritas Institute of Higher Education, Hong Kong
dDepartment of Mathematics and Information Technology, The Education University of Hong Kong, Hong Kong
*tlwong@ieee.org

Abstract: Designing a good curriculum or an appropriate learning path for learners is challenging because it requires a very good and clear understanding of the subjects concerned as well as many other factors. One common objective of educational data mining and learning analytics is to assist learners to enhance their learning via the discovery of interesting and useful patterns from learning data. We have recently developed a technique called skill2vec, which utilizes an artificial neural network to automatically identify the relationship between skills from learning data. The outcome of skill2vec can help instructors, course planners and learners to have a more objective and data-informed decision making. Skill2vec transforms a skill to a vector in a new vector space by considering the contextual skills. Such a transformation, called embedding, allows the discovery of relevant skills that may be implicit. We conducted experiments on two real-world datasets collected from an online intelligent tutoring system. The results show that the outcome of skill2vec is consistent and reliable.

Keywords: skill relationship, neural network, learning analytics, artificial intelligence

1. Introduction

Designing a suitable learning sequence or learning path is essential for learners to effectively and efficiently acquire multiple knowledge or skills (Chen, 2008). Taking learning mathematics as an example, it is natural for learners to learn in the order of addition, subtraction, multiplication and then division due to the relationship between these concepts and the skills involved. Recent studies showed that learning trajectories, that is, the paths or sequences of learning activities involved to achieve a learning goal, can have significant impact to mathematics learning (Clements & Sarama, 2004; Schoenfeld, 1992). A coherent cumulative learning sequence can help learners better conceptualize the knowledge and lead to higher order thinking (Meyers & Nutly, 2009). Curriculum designers, educators and researchers have been paying great effort in reviewing the knowledge sequence of different subjects, or the curriculum across subjects in a formal learning context (Datnow, Borman & Stringfield, 2000; See, Gorard & Siddiqui, 2017). For example, the “Core Knowledge Sequence” adopted in the United States “identifies and describes the specific core of shared knowledge that all children should learn in U.S. schools. The Sequence is intended to help children establish strong foundations of knowledge grade by grade from preschool through grade eight.” In essence, a curriculum defines the sequence of knowledge and skill learning.

In a formal learning context, teachers can schedule the sequence of knowledge or skills to be learned by students based on the curriculum designed by the authority (such as state government). Teachers may also be familiar with students’ background and capability to plan and refine the schedule of teaching based on the students’ ability to assist students to learn more efficiently and effectively. However, teaching experience is subjective and various factors, including the groups of students taught, can affect it. Therefore, teachers may have different perspectives and preferences regarding the learning sequence for different skills. Taking mathematics as an example, factors such as culture and gender differences may influence the learning sequence of different topics and, hence, the curriculum design
(Burny, Valcke, Desoete & Van Luit, 2013; Hyde & Mertz, 2009). This can pose additional difficulties in planning the learning sequence. It may also pose difficulties in planning a cross-subject curriculum, which involves the challenge of understanding the details of different subjects.

Thanks to the rapid development in information technology, a massive amount of data can be readily collected for analysis. For example, many universities and schools use course management systems such as Blackboard, moodle, and schoology for sharing learning resources, conducting online activities, attempting online tests, etc. These course management systems are able to keep track of the learners’ online activities for analysis and better understanding their learning (Romero, Ventura & Garcia, 2008). Similarly, many learning data such as the duration that the learner spends on viewing a video can be collected from massive open online courses (MOOCs). These learning data are very useful and valuable for course planners, lecturers, instruction designers, and learners themselves to enhance learning. Recently, Learning Analytics (LA) have been intensively researched to analyze the data about the learners and their contexts. The well-adopted definition of learning analytics is “the measurement, collection, analysis and reporting of data about learners and their contexts, for purposes of understanding and optimizing learning and the environments in which it occurs.” (LAK, 2011) LA can provide instructors and learners with analytical information of the latter’s learning progress and hence lead to data-driven and well-informed decisions to optimize the learning.

This paper follows up the preliminary study reported in (Wong, Xie, Wang, Poon & Zou, 2017) and investigates the application of learning analytics technique to identify the relationship between the knowledge or skills based on the learning data available. The relationship between different skills can be readily visualized and easily interpreted by human. We carefully verified our approach by conducting experiments using two real-world datasets of learning algebra in an online intelligent tutoring system (ITS). The rationale of our approach is to automatically analyze the sequences of tasks that students undertake. Tackling each task requires the application of a skill. Therefore, a task sequence can also be interpreted as an evidence of the application of a sequence of skills. We aim at identifying the relationship between different skills from these sequences. To achieve this aim, we have developed an approach called skill2vec based on artificial neural network. In essence, our approach utilizes a two-layer neural network to transform an input skill to its contextual skills. In our approach, contextual skills refer to the neighbours of a skill in the sequence. The parameters of the neural network are “trained” automatically based on the available data. After training, our approach can essentially find a mapping from a skill to a vector in a new vector space that preserves the structure between different skills. This mapping, called an “embedding,” allows the identification of the relationship between skills by projecting the skills onto the new vector space. The results of an embedding can also be visualized by further projecting the vectors in the new space onto a two-dimensional space. We conducted experiments on datasets concerning student performance on mathematical problems from an intelligent tutoring system (Stamper, Niculescu-Mizil, Ritter, Gordon & Koedinger, 2010). For example, we discovered from the data that learning “Consolidate variables” then “Combine like-terms” in solving equation was similar to learning “Correctly placing points” then “Setting the y-intercept” in drawing an equation in a coordinate system.

2. Methodology

2.1 Problem Definition

The aim of this work is to identify the relation between skills by projecting a skill onto a lower dimensional space through embedding. Suppose there is a sequence of skills denoted by \((S_1, S_2, \ldots, S_T)\), where \(S_t\) is the skill required to tackling a particular task in the \(t\)-th step and \(T\) is the length of the sequence. For example, Figure 1 shows a sample question from an online assessment system. This question lists the three steps involved in answering the question, as follows: 1. \([Triangle perimeter]\) finding the perimeter of the triangle in terms of \(k\); 2. \([Consolidate variables]\) identifying terms with same variables; 3. \([Combine like-terms]\) adding up the terms with \(k\). Each step is associated with a skill (or knowledge component), which refers to the information that can be used to accomplish the task. In this example, the length of the skill sequence is three and it consists of three different skills. Suppose there are altogether \(N\) different possible skills. Then each skill \(S_i\) can be
represented by a one-hot vector of length $N$, which contains only one element with value 1 and other elements with value 0. The vector is too sparse and does not have sufficient information from computing similarity between skills in this original vector space. To address this difficulty, the objective of our approach is to find a transformation (that is, an embedding) of this vector from the original $N$-dimensional space to a new $H$-dimensional space where $H \ll N$. The new space should be more expressive and be able to capture the context of the skills such that the relevant skills are close to each other in the embedding.

![Diagram](image)

**Figure 1.** A sample question in the dataset extracted from www.assistments.org.

### 2.2 Skill2vec

Our technique, *skill2vec*, borrows the idea of the SkipGram model, which achieves promising results in Natural Language Processing (NLP) problems (Mikolov, Sutskever, Chen, Corrado & Dean, 2013). One sample application is analogy reasoning, which can accurately infer professor given university, doctor given hospital, etc. Figure 2 depicts an overview of our *skill2vec* model. Essentially, it is a fully connected neural network model consisting of $H$ neurons in the hidden layer. The output layer contains a number of neurons depending on the predefined window size $W$, which is equal to 2 in our example.

![Diagram](image)

**Figure 2.** The *skill2vec* model.

The rationale of *skill2vec* is that for each skill $S_t$ in the sequence, the contextual skills are defined as skills for solving the neighboring tasks within a window of size $W$. The architecture of the neural network outputs the values corresponding to the contextual skills by inputting the skill $S_t$. In this way, the $N$-dimensional one-hot vector, $S_t$, which represents the skills needed to tackle step $t$, serves as the input of the *skill2vec*. This input vector will be fed to the hidden layer. The input to the $j$-th neuron of the hidden layer is defined as a standard weighted sum in the following manner:

$$ h_j = \sum_{i=1}^{[N]} x_i w_{i,j} $$
where $x_i$ and $w_{i,j}$ refer to the input value of the $i$-th element of $S_i$ and the weight connecting from the $i$-th element of $S_i$ to the $j$-th neuron of the hidden layer. The superscript of $w$ denotes that it is the weight to the first (hidden) layer of the neural network. The output layer consists of neurons that, together, show how likely it is for a particular skill to be among the contextual skills of $S_i$. The output value of each output neuron is computed using the softmax function as follows:

$$u_k = \sum_{j=1}^{H} h_j w_{j,k}^2, \quad y_k = \frac{\exp(u_k)}{\sum_{k'=1}^{S} \exp(u_{k'})}$$

where $y_k$ is the output value of the $k$-th neuron of the output layer and $w_{j,k}^2$ is the weight connecting from the $j$-th neuron of the hidden layer to the $k$-th neuron of the output layer. Essentially, the values of the output neurons represent the probability that a particular skill is the contextual skill of the input skill. On the other hand, the hidden layer of skill2vec serves as an embedding, which projects an input skill onto a new vector space in $\mathbb{R}^H$. Technically, $h \in \mathbb{R}^H$ is the product of the input vector $x \in \mathbb{R}^N$ and the weight $w^1 \in \mathbb{R}^{N \times H}$. As a result, two different skills can map to two similar vectors in $\mathbb{R}^H$ if they have similar contextual skills.

![Figure 3](image-url) **Figure 3.** The generation of training examples from a sequence of skills. The shaded and unshaded skills refer to the input and output of our skill2vec model during training.

The weight of the parameters of skill2vec can be trained or tuned automatically from the available sequences collected. For each available sequence of skills, training examples are created to train skill2vec. Figure 3 shows an example of the generation of training examples from a sequence containing eight skills. Let the window size be 2. For each of the skills in the sequence, a training example will be created along with the neighbouring skills within the window size as illustrated in the Figure. Auxiliary skills will be created for the skills at the beginning or at the end to complete a training example. With respect to the Figure, the shaded skill and unshaded skills in each training example become the input and the output of skill2vec during training respectively. Note that each skill is an $N$-dimensional vector. Given an input skill, the difference (or error) between the predicted output by skill2vec model and the actual value of the output will be measured. The weights (i.e., $w^1$ and $w^2$) of skill2vec are then updated using standard backward propagation and delta rule. This process will be repeated by using the next available training examples and until the total error among all training examples is minimized or below a certain predefined value.

2.3 Experiments

We conducted experiments to demonstrate the effectiveness of our proposed method. First, we collected a publicly available dataset that included the interaction of 1,840 and 575 students in an ITS when they learned algebra in the academic years 2005–2006 and 2006–2007, respectively (Stamper, Niculescu-Mizil, Ritter, Gordon & Koedinger, 2010). We call these two datasets Algebra-2005-2006 and Algebra-2006-2007, respectively. In the ITS, a learner performed a sequence of tasks to answer a given question similar to the sample question in Figure 1. The total numbers of steps for all students
involved in Algebra-2005-2006 and Algebra-2006-2007 are 813,661 and 2,289,726, respectively.

We pre-processed the datasets by forming sequences of skills; each sequence being generated from a single question. Training examples were then generated from these sequences as described in Section 2.2. Training examples from these two datasets were used for training two skill2vec models separately. These two models, one from Algebra-2005-2006 and one from Algebra-2006-2007, served to triangulate the results of our model. The vector corresponding to each skill transformed to the new vector space by the embedding was then computed as described in Section 2.2. Next, the skill vectors in the new vector space were projected onto a two-dimensional space for visualization and analysis.

3. Results

Figures 4 and 5 show the visualization of parts of the results of our experiments on the datasets Algebra-2005-2006 and Algebra-2006-2007 respectively. The actual output of skill2vec contains hundreds of vectors related to all skills. For simplicity of exposition here, we only select and display 8 different related skills in Figures 4 and 5 to illustrate the usefulness and effectiveness of skill2vec. As described in Section 2.3, the vectors corresponding to the skills were projected on a two-dimensional space for better visualization. In each of the Figures, a point refers to a particular vector and its label shows the skill label associated with the vector. The arrows were added manually to connect the relevant skills in the results. For example, the skills “S1: Consolidate variables” and “S2: Combine like-terms” refer to two different, but related skills in algebra.

![Figure 4](image1.png)  ![Figure 5](image2.png)

Figure 4. The results of applying skill2vec to the dataset Algebra-2005-2006.

Figure 5. The results of applying skill2vec to the dataset Algebra-2006-2007.

We found that the same skills are located in similar positions in Figures 4 and 5. The differences (or directions) between two different related skills are also similar. For example, “S2: Combine like-terms” is located at the southeast of “S1: Consolidate variables”, and “S3: Identify units” is also located at the southeast of “S4: Changing axis bounds” in both figures. However, “S5: Entering the slope” and “S6: Variable in denominator” are in the middle of Figure 4, but they are at the top of Figure 5. Recall that the two skill2vec models were trained using two different datasets Algebra-2005-2006 and Algebra-2006-2007 and the training involved randomness in neural network initialization. Hence, the two embeddings (and the new vector spaces) would be expected to be different unless they have some intrinsic relations. Nevertheless, the vectors in both embeddings and the difference between related vectors are largely consistent.

4. Conclusion

We have developed an approach called skill2vec for automatically identifying skill relationship from learning data. Skill2vec utilizes a two-layer neural network architecture whose input and output are a skill in a sequence and its neighbouring skills respectively. Skill2vec can find an embedding, which is a mapping transforming a skill from its original vector space to a new vector space. The new vector space is more expressive and related skills can be identified by considering the vectors in this new
space. We conducted experiments on two different real-world datasets collected from an online intelligent tutoring system to demonstrate the effectiveness and reliability of our approach.

Acknowledgements

The work described in this paper is fully supported by the grant from Research Grants Council of the HKSAR (Ref.: UGC/FDS11/E02/15).

References


