# Transfer Learning in Intelligent Tutoring Systems — Results, Challenges, and New Directions

**Aubrey Gress, Ian Davidson**
University of California, Davis
Davis, California 95616

**J. T. Folsom-Kovarik**
Soar Technology, Inc.
Ann Arbor, Michigan 48105

**Abstract**

At the core of an intelligent tutoring system is the ability to estimate a student's level of skill proficiency. However, making accurate skill estimates can require asking the student relatively many questions. We address this challenge by using "transfer learning," a field of machine learning which uses data from related, but different, "source" domains to aid in learning in a poorly labeled "target" domain. Thus, to predict the skill of a student who hasn't answered many "target" skill questions, we use estimates of well tested "source" skills. We explore settings where the student has answered no questions related to the target skill (the cold start setting) and those where she has answered a few (the warm start setting). We focus on the challenging situation where the domain expert has **not** identified the relationship between the skills. We find that the Ridge estimator is useful for transferring knowledge from source to target skills, outperforming nonparametric regression methods and a baseline which only uses student performance on target skill questions.

## 1 Introduction

Intelligent Tutoring Systems (ITS) are systems which simultaneously estimate and improve upon the skills of students (Desmarais and Baker 2012). For example, an ITS to teach children arithmetic may present a student with a series of questions in order to estimate the students' abilities in four separate areas - addition, subtraction, multiplication and division - and adaptively present lessons and questions in order to improve the assessment of student's arithmetic skills in the four areas.

Accurately estimating skill level when a student has answered few or no questions in an area is a challenging problem for ITS. Standard methods address these "warm-start" and "cold-start" problems by modeling the relationship between skills using a Bayesian Network with meaningful network topology and informative priors representing skill estimates, but this requires a domain expert to manually construct the network. This can be both expensive and error-prone if the domain expert makes a judgment mistake. Furthermore, the priors are likely to reflect average student performance, making them detrimental for use in any atypical student situations.



We take a different approach by modeling the cold-start problem as a supervised transfer learning problem, using competency estimates from previous "source" skills to estimate a "target" skill. Our method does not need a domain expert to model the relationship between skills. Rather, the relationship between source and target skills is automatically learned. This is a challenging problem particularly when there are many source skills to transfer, but can be solved using supervised machine learning methods.

Our paper has the following outline. We first describe a model of the ITS data we use. This model is a superset of the data required to perform our experiments, but allows us to better discuss future work. We then sketch previous work and our own approaches before presenting experimental results, future directions and concluding.

## 2 Data Model of a Student

Here we overview an idealized data set. Not all of this data was available for our experimental results, but we include a full description to better describe our future work and challenges. At a high level, the data used in ITS can be modeled as a series of transactions between different types of "objects" of interest. Specifically, we assume the existence of four types of objects:

- Users: students, trainees, or other learners who interact.
- Skills: skills associated with users, such as arithmetic skills like addition and division. Objects of this type may also represent concepts to learn, attitudes to impart, and so on.
- Questions: questions which users interact with in some way that gives insight to the user's skills, such as questions on a test or homework. These may also represent test items, observed performance in a simulation, and so on.
- Study Materials: objects users can interact with, but which don't necessarily measure a user's competencies, such as a study guide or a lecture video.

Assuming we have sets $S = \{S_1, ..., S_s\}$ of users, $Q = \{Q_1, ..., Q_q\}$ of questions, $K = \{K_1, ..., K_k\}$ of skills and $M = \{M_1, .., M_m\}$ of study materials, a common set of transactions/interactions are:

| Question ID | Skill |
|---|---|
| Q1 | Addition |
| Q2 | Addition |
| Q3 | Multiplication |
| Q4 | Multiplication |
| Q5 | Division |
| Q6 | Division |

Table 1: An example of information available for questions in the data sets we used. Notably, each question is only associated with a skill.

| Student ID | Question ID | Score |
|---|---|---|
| S1 | Q1 | 1 |
| S1 | Q2 | 1 |
| S1 | Q3 | 0 |
| S1 | Q4 | 0 |
| S2 | Q1 | 1 |
| S2 | Q5 | 1 |
| S3 | Q2 | 1 |
| S3 | Q3 | 0 |

Table 2: An example of information we used for student-question transactions in our experiments. Score is binary - 1 for correct, 0 for incorrect.

- User-Question: the results of the interaction between a user $S_i$ and an item $Q_j$. For example, if user $S_i$ has answered $Q_j$. This transaction would also store when this annotation took place and could store how long it took the user to make the annotation.
- User-Skill: a users skill value, such as their skill in arithmetic. This would also have a time component, to model that user skills may change over time.
- Skill-Question: skills associated with the item, such as skills tested by an exam question.
- Skill-Skill: the relationship between two skills, such as their similarity, or prerequisite relations.

Other forms of pairwise data could be available as well. Importantly, some of the data may be missing, such as only having the interactions between user $S_i$ and a subset of the questions. This represents an ITS providing useful adaptation to individual learners without needing to fully test every skill. Also, while it's generally assumed that the time of any transaction involving a user will be stored, a time component could be associated with any object or pairwise transaction to model change over time.

Additionally, other features may be available for some of the objects. For example, $S_i(x)$ could include not just skills but also demographics information for user $i$.

Due to data set limitations, in our experiments the only data we used was the set of questions and student performance on questions. For illustrative purposes, synthetic examples of the data we used are presented in tables 1 and 2.

# 3 Previous Work

ITS generally estimate students skills and update these estimates based on question performance. These systems provide questions in an adaptive manner in order to more accurately estimate the student's skill and to better train the student in these skills (Beck, Stern, and Woolf 1997), (Falmagne et al. 2006), (Desmarais and Baker 2012). However, because each students' performance is considered independently, systems may need to ask relatively many questions in order to accurately estimate student skills.

Bayesian Networks can be used to to reduce the number of questions that need to be asked. This is particularly common when the system models multiple skills. By using Bayesian Networks, performance on questions about one skill can allow inference about the status of other skills. For example, if a student answers a question on multiplication correctly, then they have likely mastered addition. While modeling skill relationships can lead to more accurate estimates, it requires a domain expert to construct the network (Koppen and Doignon 1990). Additionally, there is still the problem of accurate skill estimation when very few questions have been asked.

Transfer Learning, in the machine learning sense, is the problem of learning a function when there exists some discrepancy in the generating distributions of the train and test sets (Pan and Yang 2010). Here, the training set and test sets are called the "source" and "target" data sets. Generally, the source data will be drawn from a problem that is related, but different from the target data and the challenge is to optimize performance on the target by somehow overcoming the differences between the data sets. For example, if the task is to predict the outcome of a new medicine, the source data may consist of patients over the age of 50 while the target data only contains patents under 40. These two data sets are similar because they both model the success of the drug, but they differ in the demographics of the patients, which are expected to affect the drug's success in unpredicted ways. As such, learning algorithms agnostic to this shift may perform poorly on the target data.

Transfer learning algorithms generally vary in *what* they transfer and *how* they transfer. For example, one class of transfer learning algorithms assign higher "weight" to source instances believed to be most similar to the target set (Ben-David et al. 2007). Conversely, other algorithms use hypotheses learned from the source data to regularize the learning of a hypothesis on the target data (Tommasi, Orabona, and Caputo 2010).

The class of transfer learning algorithms our work focuses on is a form of "Feature Representation" transfer, wherein predictions made by hypotheses trained on the source data are used as features for the target data (Jie, Tommasi, and Caputo 2011). These methods first, for each source data set, independently learn a hypothesis then augment the feature representation of target instances with predictions made by be these hypotheses. For example, if the original feature representation of a target instance is $x = [x_1, ..., x_p]$ and there are $k$ sources, a hypothesis $f_i$ will be learned for each source and the final target feature representation will be $[x_1, ..., x_p, f_1(x), ..., f_k(x)]$. Within the context of ITS,

this means using skill estimates of a set of source skills as features for the target skill prediction problem.

## 4 Our work

We are interested in the problem of predicting the competency of a student in some skill $s$ before they've answered many or even *any* questions that test the skill. These "warm-start" and "cold-start" problems are important because accurate initial competency estimates can lead to better use of learning time through well-aligned question and topic allocation from the start. Also, this problem arises when trying to decide who, among a set of candidates, should receive further training in a new area.

Given a rich feature representation for a student, skill estimation can be solved as a standard supervised machine learning problem by providing a training set $\{(x_1, y_1), ...(x_n, y_n)\}$, where $x_i$ is the feature representation and $y_i$ is the skill value of student $i$. While generating skill estimates $y_i$ for a training set is possible by simply having a set of students answer questions that test skill $s$, generating a feature representation $x_i$ is much trickier. Demographics information could be used, but won't be available in many settings and may not be useful for competency prediction.

To solve the feature representation problem we assume the existence of some set of "source" skills for which we have competency estimates. The underlying assumption of our work is that skill estimates from these source skills can be used to better estimate the "target" skill. For example, we can assume that given estimates of a students abilities in addition and subtraction, this information can be used to make better estimates of their multiplication ability.

To solve this problem we assume we have accurate skill estimates for both the source and target skills for some subset of students. We then transform this into a machine learning problem by using the source skill estimates as features and the target skill estimate as the label. For example, if $x^i_{ST} = [x^i_{S_1} x^i_{S_2} ... x^i_{S_p} x^i_T]$ are student $i$'s skill estimates, where $S_i$ is the i$^{th}$ source skill and $T$ is the target skill, we split up the source and target estimates and use $x^i_S = [x^i_{S_1} x^i_{S_2} ... x^i_{S_p}]$ as the student's feature representation and $x^i_T$ as the label. Finally, we apply a standard supervised machine learning algorithm to learn a mapping from the feature set to label.

**Negative Results.** We initially experimented with two nonparametric methods: the Nadaraya-Watson estimator (Friedman, Hastie, and Tibshirani 2001) and Learning with Local and Global Consistency (Zhou et al. 2004), both using Euclidean distance. These methods both performed poorly. We believe this is because both algorithms require some measure of distance between pairs of instances, but the Euclidean distance assumes all features are equally important. Within the context of skill transfer, it seems unlikely that all skills would be equally predictive for all other skills. Furthermore, nonparametric methods perform worse in high dimensional space due to the "curse of dimensionality" (Friedman, Hastie, and Tibshirani 2001) and in our experiments we explore transferring more than 20 source skills. There are methods for learning a metric (Xing et al. 2003) which may help solve these problems, but these methods require solving computationally expensive optimization problems and can require large training sets.

Instead, in our experiments presented here, we used the Ridge estimator (Friedman, Hastie, and Tibshirani 2001) which learns a linear function by solving the following optimization problem:

$$\min_{w,b} \sum_i (x^i_S w + b - x^i_T)^2 + \lambda ||w||^2 \quad (1)$$

where $\lambda \geq 0$ is a regularization parameter. Experimentally, we found this method works well because it's able to assign higher weights to features which are more predictive. However, it may be interesting to consider different estimators for future work.

**Ridge vs Lasso.** We chose not to use the Lasso (Tibshirani 1996), which is a modification of the Ridge which uses the $\ell_1$ norm in order to promote sparsity of $w$. While the Lasso is useful when the goal is to learn the "correct" function or a more easily interpreted function, our goal is prediction accuracy. Furthermore, the sparsity property of the Lasso implicitly makes the assumption that only a fraction of the source skills are relevant. This is a strong assumption that may not hold in practice, and the Ridge does not make such an assumption

Given the function estimate produced by the Ridge, we can estimate the target skill for new students, even if they haven't answered any questions pertaining to skill $T$, as long as they have answered questions that test the source skills. For example, to predict a student's skill in multiplication, we can use the student's skills in addition and subtraction. This is the "cold start" problem.

We also studied the "warm start" setting, where in addition to source skill estimates students have answered a small number of target skill questions. For this setting we combined the source-to-target skill estimate $\hat{f}_{ST}$ with an estimate $\hat{f}_T$ derived using only target skill questions. Specifically, the final prediction is $\alpha \hat{f}_{ST} + (1 - \alpha)\hat{f}_T$ where $\alpha \in [0, 1]$ is a parameter that is tuned using standard model selection techniques. We found that generating predictions using both these sources of information can lead to even more accurate predictions.

In summary, traditional methods using Bayesian Networks transfer knowledge of student competencies between skills as well, but the topologies of these networks must be manually constructed by domain experts. Additionally, even if a topology is available, optimizing parameters in a Bayesian Network solves a fundamentally different inferential problem than we face in the cold and warm start problems. Specifically, algorithms to tune Bayesian Networks optimize a probability distribution that minimizes some notion of error with the full joint probability distribution, while the goal of our problem is to minimize the error in predicting competencies of a specific skill in the warm and cold start settings. The latter goal is directly optimized by supervised learning methods such as the Ridge estimator.

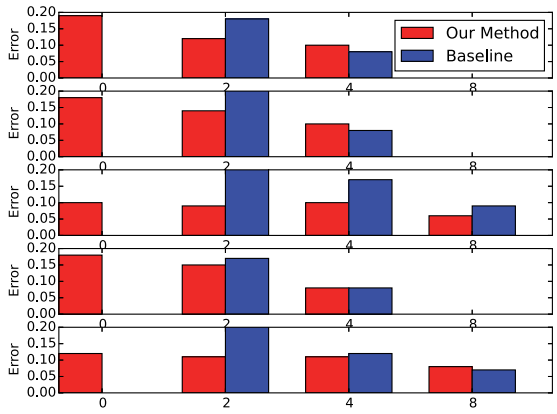


Figure 1: Error in skill prediction on "Digital Games for Improving Number Sense" data. x-axis is the number of target questions available. Note that 0 indicates the cold start setting. Our method performs much better than the baseline when few questions are available. Missing bars indicates not enough data was available to run the experiment. Skills are: all, all*NearBench, all*NotNear, all*NotNear*Click, all*NotNear*Click*NonUnitNotNear

## 5 Experiments

For our experiments we used the "Digital Games for Improving Number Sense" and "USNA Physics Fall 2008" (Vanlehn et al. 2005) data sets from the LearnSphere data repository (Koedinger et al. 2010). The Number Sense data set records performance of 51 elementary school students on fraction and decimal visualization tasks. The USNA Physics data set records performance of 69 university students using the Andes physics tutoring system. To estimate each student's skill we averaged their performance for each skill separately. For example, if the student correctly answered 3 out of 5 questions for skill $i$ then we recorded their skill as .6.

We compared our method in both the cold and warm start settings to a baseline method. This baseline returns the average performance of the student on all available target questions. Thus, estimates of source skills are unused. Importantly, this baseline cannot be used in the cold start setting since there are no target questions answered to estimate the target skill proficiency.

Figures 1 and 2 are our results for our method compared to baseline results. These results show our method significantly outperforms the baseline when few target questions are available. Additionally, our method can make accurate predictions in the cold start setting. When many target questions are available, our method generally performs comparably to the baseline.

## 6 Future Directions on Transfer Learning and ITS

Here we describe a variety of future directions which will lead to more informative and efficient transfer learning which in turn will result in more accurate skill estimates using fewer answered questions.

### 6.1 Learning Skill Models and Skill Pathways

Here we envision attempting to learn the order in which skills are mastered. We can consider two data situations, the first described by the data used in our experiments to estimate skill proficiency *once* and another when skill estimates are made at regular intervals as increasingly more questions are answered.

In the first setting for each student we can create a rank ordered list (in terms of decreasing proficiency) of each each skill. We will then have a collection of such lists (one for each student) and from each list can create many pairs of the form skill $i$ is ranked above skill $j$. If all rankings are consistent across all students then finding a consensus ranking is trivial, however inconsistent rankings (skill $i$ ranked above skill $j$, skill $j$ ranked above skill $k$ and skill $k$ ranked above skill $i$) will exist. Finding a consensus ranking from potentially inconsistent individual rankings is known as the Rank Aggregation problem (Coleman and Wirth 2009). Such a consensus ranking gives a simple ordering of how the skills are most likely to be acquired for the population of students.

In the second setting since we have multiple rankings per student (at various fixed intervals). We can than create multiple consensus rankings for each time step which then shows how the rankings change over time. If skill $i$ is consistently ranked above skill $j$ we can infer that skill $i$ requires skill $j$. We can find even find more complex insights such as if skill $i$ is ranked above skill $j$ but when skill $k$ is learnt well then skill $i$ and $j$ rankings increase then skill $k$ is an enable of skills $i$ and $j$. Turning these insights into a graph is a long term aim.

A particularly challenging situation is to simultaneously perform clustering of the rankings and find a consensus ranking for each cluster. Then each consensus ranking (for each cluster) represents a student pathway through the skill set.

### 6.2 Identifying Most and Least Informative Questions

Here our aim is identify the most informative question associated with a skill to assess that particular skill. This can be extended to identifying the most informative question associated with skill $i$ which when transferred is most useful at estimating skill $j$. Similar calculations can be performed to ascertain the least informative question with the belief being that such questions are mislabeled by domain experts as testing a particular skill.

### 6.3 Incorporating Preparation Suggestions

A particular important topic is to predict what preparation (study material) would most increase a student's proficiency at a skill and a related question being to estimate how much the study material will improve skill proficiency. The former is a ranking problem and the later a regression problem. Transfer extensions to ranking and regression problems (Qian et al. 2014) can be used to understand the effect of study materials for skill $i$ on the proficiency of skill $j$.

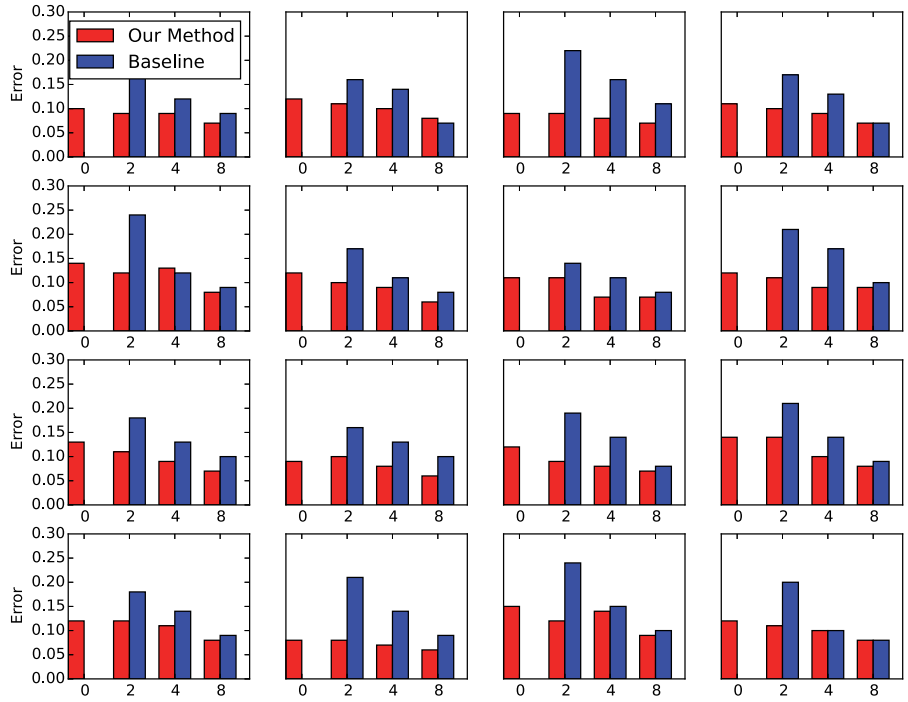


Figure 2: Error in skill prediction on "USNA Physics" data. x-axis is the number of target questions available. Note that 0 indicates the cold start setting. Our method performs much better than the baseline when few questions are available. From left to right and top to bottom, skills are: 'Angular Momentum, MOMR', 'Circular Motion, ROTS', 'Energy-Work, E', 'Fluids, FLUIDS', 'Free Body Diagrams, FBD', 'Linear Momentum, CM', 'Linear Momentum, IMP', 'Linear Momentum, LMOM', 'Power, POW', 'Rotational Dynamics, DR', 'Rotational Kinematics, KR', 'Statics, S', 'Translational Dynamics, DT', 'Translational Kinematics, KT', 'Vectors, VEC', 'Waves, WAVE'

## 7 Conclusion

The ability to estimate a student's level of skills proficiency is central to student assessment. We addressed the cold-start and warm-start problems by using transfer learning, a developing area of machine learning that transfers knowledge of a source task $i$ to learn the transfer task $j$. We showed show how transfer learning can be used to predict a target skill from a *collection* of source skills. We found that the Ridge performed better than nonparametric methods when transferring from many source skills and showed our method outperforms a baseline on two data sets from the LearnSphere repository. We outlined several future directions including creating skill pathways (the trajectories students take whilst learning a set of skills), estimating the the most informative set of questions and identifying the most useful set of study material.

## 8 Acknowledgments

This work is supported in part by the Office of Naval Research via contract N00014-15-P-1184, "DARTS-TARGET: Transfer via active requests to generalize effective training." The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Department of Defense or Office of Naval Research. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

## References

Beck, J.; Stern, M.; and Woolf, B. P. 1997. Using the student model to control problem difficulty. In *User Modeling*, 277–288. Springer.

Ben-David, S.; Blitzer, J.; Crammer, K.; Pereira, F.; et al. 2007. Analysis of representations for domain adaptation. *Advances in neural information processing systems* 19:137.

Coleman, T., and Wirth, A. 2009. Ranking tournaments: Local search and a new algorithm. *Journal of Experimental Algorithmics (JEA)* 14:6.

Desmarais, M. C., and 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.

Falmagne, J.-C.; Cosyn, E.; Doignon, J.-P.; and Thiéry, N. 2006. The assessment of knowledge, in theory and in practice. In *Formal concept analysis*. Springer. 61–79.

Friedman, J.; Hastie, T.; and Tibshirani, R. 2001. *The elements of statistical learning*, volume 1. Springer series in statistics Springer, Berlin.

Jie, L.; Tommasi, T.; and Caputo, B. 2011. Multiclass transfer learning from unconstrained priors. In *2011 International Conference on Computer Vision*, 1863–1870. IEEE.

Koedinger, K. R.; Baker, R. S.; Cunningham, K.; Skogsholm, A.; Leber, B.; and Stamper, J. 2010. A data repository for the edm community: The pslc datashop. *Handbook of educational data mining* 43.

Koppen, M., and Doignon, J.-P. 1990. How to build a knowledge space by querying an expert. *Journal of Mathematical Psychology* 34(3):311–331.

Pan, S. J., and Yang, Q. 2010. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering* 22(10):1345–1359.

Qian, B.; Wang, X.; Cao, N.; Jiang, Y.-G.; and Davidson, I. 2014. Learning multiple relative attributes with humans in the loop. *IEEE Transactions on Image Processing* 23(12):5573–5585.

Tibshirani, R. 1996. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)* 267–288.

Tommasi, T.; Orabona, F.; and Caputo, B. 2010. Safety in numbers: Learning categories from few examples with multi model knowledge transfer. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, 3081–3088. IEEE.

Vanlehn, K.; Lynch, C.; Schulze, K.; Shapiro, J. A.; Shelby, R.; Taylor, L.; Treacy, D.; Weinstein, A.; and Wintersgill, M. 2005. The andes physics tutoring system: Lessons learned. *International Journal of Artificial Intelligence in Education* 15(3):147–204.

Xing, E. P.; Ng, A. Y.; Jordan, M. I.; and Russell, S. 2003. Distance metric learning with application to clustering with side-information. *Advances in neural information processing systems* 15:505–512.

Zhou, D.; Bousquet, O.; Lal, T. N.; Weston, J.; and Schölkopf, B. 2004. Learning with local and global consistency. *Advances in neural information processing systems* 16(16):321–328.