Extending ACT-R to Tackle Deceptive Overgeneralization in Intelligent Tutoring Systems

Marshall An

Carnegie Mellon University, Pittsburgh PA, 15213, USA

Abstract

This research extends the ACT-R cognitive architecture to tackle deceptive overgeneralization within Intelligent Tutoring Systems (ITS). Existing adaptive learning technologies, while effective, rely on learning data that may not fully capture the nuances of learner understanding, particularly in cases of deceptive overgeneralization. This phenomenon occurs when learners exhibit correct actions during monitored learning sessions, yet these actions are grounded in an incomplete understanding of the necessary conditions. Due to the reliance on observed correctness, ITS may falsely assess mastery, potentially ceasing to provide further necessary practice opportunities that could aid in the refinement of understanding. This study aims to identify ITS designs that may inadvertently foster such misconceptions and to develop methods for their detection, diagnosis, and correction. Utilizing experimental designs, think-aloud protocols, and educational data mining, the research seeks to refine the adaptivity of ITS and enable more accurate assessments of true skill mastery. This work contributes to Technology-Enhanced Learning (TEL) by enhancing the precision of automated assessments and supporting more reliable adaptive learning experiences.

Keywords

Adaptive Learning, Intelligent Tutoring Systems, Instructional design, Feedback, Educational Data Mining, Bayesian Knowledge Tracing

1. Introduction

Adaptive learning technologies, powered by learning data and dynamically adjusting to individual learner needs, have proven effective across various educational settings [1]. However, by definition, any type of adaptivity relies on data reflecting student learning [1, p. 523]. The accuracy and completeness of learning data are therefore critical. There are instances, however, where the learning data may fall short, particularly in cases of *deceptive overgeneralization*.

Deceptive overgeneralization describes an undesired learning state wherein a learner acquires a relevant but incomplete subset of the conditions necessary for a skill, yet manages to perform the correct actions. Such overgeneralization is "deceptive", as it can lead to seemingly satisfactory performance during scrutinized learning sessions, as the learner's observable actions align with those of individuals who have accurately mastered the skill. However, these actions are based on a flawed understanding of the underlying conditions.

Deceptive overgeneralization poses a significant challenge, leading to false evaluations of mastery, which drives adaptivity. This can mislead learners, instructors, and researchers into getting prematurely convinced that a skill has been mastered. Many Technology-Enhanced Learning (TEL) environments, especially those utilizing Intelligent Tutoring Systems (ITS) with adaptive capabilities that dynamically select practice problems based on estimated skill mastery, might amplify the issue of deceptive overgeneralizations. Such environments may prematurely cease providing further necessary practice opportunities that aid in the refinement of understandings, leaving these inaccuracies unaddressed. The consequences of failing to detect and address deceptive overgeneralizations can extend beyond academic performance, potentially affecting long-term educational pathways, career trajectories, and in some cases, leading to dire consequences.

My doctoral research aims to investigate the mechanisms of deceptive overgeneralization by applying and extending the well-established cognitive architecture, Adaptive Control of Thought – Rational (ACT-R) [2, 3]. This study aims to uncover how certain designs of ITS might overlook subtle instances of deceptive overgeneralization and to investigate design principles that can detect and remedy them. Ultimately, my research seeks to contribute to the advancements of adaptive learning technologies, enhancing their effectiveness as educational solutions.

Proceedings of the Doctoral Consortium of the 19th European Conference on Technology Enhanced Learning, 16th September 2024, Krems an der Donau, Austria

haokanga@andrew.cmu.edu (M. An)
0009-0005-5165-640X (M. An)

2. Literature Review

2.1. Adaptive Control of Thought – Rational (ACT-R)

ACT-R, a cognitive architecture for understanding and modeling human cognitive processes, posits that cognitive behaviors are orchestrated by productions [4, 3]. A production can be represented as a condition-action pair [2, p.5], with the *condition* part specifies the circumstances under which the production can apply, and the *action* part specifies what should be done when production applies [5, p.3]. ACT-R has significantly influenced the development of ITS, which delivers personalized tutoring by adapting to the unique learning needs of each learner. Empirical studies underpinning ACT-R have led to a proliferation of ITS that successfully enhance learning outcomes across diverse educational settings [6, 7, 8]. These systems, particularly cognitive tutors, require the development and integration of domain-specific cognitive models that adhere to the ACT-R framework, to capture various learner strategies and potential misconceptions.

2.2. Adaptivity of Intelligent Tutoring Systems

Adaptive learning, fundamental to ITS efficacy, is supported by various theoretical perspectives, such as Vygotsky's zone of proximal development [9], the cognitive apprenticeship model [10], the expertise reversal effect [11], and the assistance dilemma [12]. The efficacy of ITS in improving learning outcomes is largely attributable to its adaptivity, which allows for personalized learning based on individual learner progress and needs.

Adaptivity is not a binary property, but rather "a matter of degree" [1, p.523]. ITS distinguish themselves by adapting across all three major time scales defined by the Adaptivity Grid: step, task, and design [1, p.525].

Within the step-loop, ITS provides timely and targeted feedback at each problem-solving step. Indeed, timely feedback is critical to enable the learners to continuously monitor their learning and evaluate their problem-solving strategies and their current understanding [13]. The positive effects of feedback are well supported by the rich wealth of evidence in the literature review by Shute [14]. Feedback is most effective when it clearly highlights discrepancies between a learner's current performance and the desired outcome, while offering actionable guidance to help learners meet specific target criteria [15, p.139]. ITS embody these best practices of feedback, by detecting and diagnosing observable discrepancies between expected and actual actions at each step. With a developed cognitive model, a cognitive tutor employs *model tracing* to compare learner actions at each problem-solving step against the possible actions generated by the cognitive model, in order to provide individualized, just-in-time learning support tailored to the learners' specific approach to a problem [16, p.142].

In the task-loop, ITS employ *knowledge tracing* algorithms such as Bayesian Knowledge Tracing (BKT) [17] to dynamically adjust problem sequences based on real-time assessments of learner mastery. Each time a learner attempts a step in a practice problem, the system updates its estimate of the learner's mastery of the relevant production rule based on the correctness of the learner's action [16, p.143]. This ongoing assessment allows ITS to dynamically tailor the sequence of problems, ensuring that each practice opportunity aligns with the learner's current skill level and learning trajectory. When the system reaches a high degree of certainty, typically exceeding a predefined threshold (e.g., 95%) [16, p.144], about a student's mastery of a skill through repeated observations of correct actions, it ceases presenting tasks related to that skill. This automated stopping rule optimizes the balance between learning time and effort, preventing overpractice and maximizing educational efficiency.

Furthermore, the design-loop adaptivity involves data-driven instructional (re)design, before and between iterations of ITS development, informed by learning data [1, p.526].

However, the adaptivity of ITS is not without limitations. One key challenge lies in addressing deceptive overgeneralization—where learners perform correct actions based on a flawed understanding of underlying conditions. This phenomenon challenges the assessment models of ITS, which typically rely on differentiating between correct versus incorrect actions to gauge mastery. As such, deceptive overgeneralization presents an intriguing area for further research.

3. *Deceptive Overgeneralization* as a Possible Learning State

Learning is typically characterized by a gradual and continuous process rather than sudden transformative insights [18]. The Knowledge-Learning-Instruction (KLI) framework views learning as the acquisition of Knowledge Components (KCs), which are acquired units of cognitive functions or structures [19]. The KLI framework identifies *induction and refinement* as one primary type of learning processes, particularly for acquiring KCs associated with variable conditions: for KCs with conditions that can vary in form or value, learners must induce and refine KCs so that the acquired KCs are "accurate, appropriately general, and discriminating" [19]. As we consider the induction and subsequent refinement of a KC as a continuous learning progression, learners may initially acquire an inaccurately generalized version of

Table 1
Examples of Correct and Inaccurate Generalization in Knowledge Components Across Various Disciplines

Discipline / Topic	Correct KC	Referenced Inaccurate KC
Math / Geometry	IF the triangle is isosceles AND two angles are at the base of the triangle THEN the two angles are equal	IF the triangle is isosceles AND two angles THEN the two angles are equal [20]
Language / English Articles	IF single mountain name THEN zero article	IF mountain name THEN zero article [21]
Statistics / Data Visualization	IF categorical data THEN choose pie chart	IF demographic data THEN choose pie chart [22]

the target KC. This initial misunderstanding may either be refined into an accurate KC through further practice, or it may persist as inaccurate due to a lack of practice opportunities that support the refinement process.

3.1. Modeling of Deceptive Overgeneralization

A KC connects features of a problem to a corresponding response. A learner has acquired a KC that is considered accurate, or "with high feature validity", when all of the features are relevant to making the response and none of them are irrelevant [23]; otherwise, a KC is inaccurate and requires further refinement. Inaccurate generalization could be overgeneralization, undergeneralization, or even more nuanced a mix of them. Indeed, inaccurate generalization is a common phenomenon observed in learning sciences research across various disciplines. Table 1 presents examples of incorrect generalization, along with their corresponding accurate KCs, drawn from research literature. Among these, deceptive overgeneralization is particularly intriguing to investigate.

In ITS, specifically those developed using Cognitive Tutor Authoring Tools (CTAT) [24, 25], each production's condition-action pair is structured as an IF-THEN statement [26]: `IF <condition> THEN <action>`. Overgeneralization occurs when a learner acquires production rules whose IF part is overly broad compared to the correct IF part. In computational or logical terms, overgeneralization can happen due to the omission of logical AND operators in the IF part. Consider a target KC requiring multiple conditions for its activation, represented as `IF A AND B THEN <action>`. Overgeneralization might arise when a learner acquires a KC that omits part of the conditions, resulting in `IF A THEN <action>`.

It is crucial to distinguish the phenomenon of deceptive overgeneralization from the broader concept of "misconceptions." Consider a simple algebra problem: Anderson describes an observation that a student incorrectly solves the equation $2x = 6$ by subtracting 2 from both sides, erroneously resulting in $x = 4$ instead of $x = 3$ [18]. Such misconceptions lead to actions that are clearly incorrect, allowing for immediate observation, feedback provision, and tailored subsequent training. In contrast, deceptive overgeneralization involves learners who, during closely monitored learning sessions, apply correct actions that are based on incomplete understanding of the necessary conditions. These learners may later inappropriately apply these actions under unsuitable circumstances, often beyond the scrutiny of the initial learning. This highlights why deceptive overgeneralization is particularly "deceptive": learners are still observed to take correct actions, despite their misconceptions.

Furthermore, my research differs from prior studies that have primarily focused on distinguishing between superficial and deep features in learning. Superficial features, also known as shallow or surface features, are those that do not contribute to correct solution pathways [22, 27, 28]. For example, a learner chose to use a pie chart because the data is demographic (superficial) rather than categorical (deep) [22]. In contrast, my research investigates scenarios in which learners take correct actions based on a relevant yet incomplete set of features. Importantly, unlike superficial features, all these features belong to the correct solution pathways, thereby making the learners' understanding appear deceptively correct.

3.2. Stickiness of Deceptive Overgeneralization

The KLI framework delineates a relationship between observable and unobservable events: instructional events, learning events, and assessment events [19]. Instructional events cause learning events, which are unobservable processes that result in changes in KCs, such as acquisition of new KCs or refinement of existing KCs. The changes of KCs, in turn, cause learner performances that are observable during assessment events. Given that learning events are central yet unobservable, assessments are expected to be designed with the quality to accurately reflect the true nature of learning events. However, in cases of overgeneralization, certain designs may fail short. Using set theory, overgeneralization can be visualized as an inclusion relation and we can identify a specific type of potential design flaw, as depicted in Figure 1.

Many TEL environments, particularly those involving ITS, leverage automated evaluation and feedback mechanisms to deliver learning at scale. The reliance

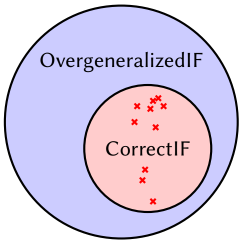


Figure 1: Overgeneralization occurs when a learner acquires production rules whose IF part is a superset of the correct rule's IF part, covering an overly extended range. This relationship can be expressed as OvergeneralizedIF ⊇ CorrectIF. Cross marks within CorrectIF represent practice activities that cannot test for overgeneralization. If all practice activities fall within CorrectIF, focusing solely on correct actions, the instructional design will fail to identify whether learners have acquired the correct rule or an overgeneralization.

on these automated mechanisms can pose challenges for all stakeholders regarding deceptive overgeneralization. TEL tools might mistakenly provide positive feedback to learners who perform correct actions based on an inaccurate understanding of conditions, inadvertently reinforcing misconceptions. Instructors and researchers employing learning analytics or educational data mining are similarly at risk of being misled by seemingly satisfactory learning data, potentially missing opportunities for intervention and correction that address learners' incorrect understandings. Moreover, ITS, with its adaptive capabilities that dynamically select practice problems and assess mastery, might amplify these issues. The reliance on observed correctness by knowledge tracing algorithms can lead to premature conclusions about learner mastery, halting further necessary practice that aids genuine skill development and refinement, leaving those misconceptions unaddressed. As what is captured and reported by TEL tools appears correct, encouraging, and satisfactory, deceptive overgeneralization may be particularly "sticky" and resistant to detection and change.

3.3. Both Novices and Experts Could be Prone to Deceptive Overgeneralization

If "practice makes perfect" were true to the extent that well-developed expertise guarantee refined and accurate skills, then deceptive overgeneralization could be effectively addressed by providing ample practice opportunities in favorable learning conditions. However, I argue that even experts are not immune to deceptive overgeneralization, despite their considerable mastery of skills.

Ambrose et al. [15, p.97] modeled mastery and its development into four stages, as illustrated in Figure 2. As this model suggests, while competence develops in a more-or-less linear fashion, consciousness initially increases and then decreases, as both novices (in Stage 1) and experts (in Stage 4) operate in states of relative unconsciousness, though for vastly different reasons [15, p.97]. I contend that deceptive overgeneralization may occur during any stage transition, including transitions towards Stage 4. Experts, as they develop their proficiency and automaticity, may also be prone to forming inaccurate heuristics and cognitive shortcuts to enable fast task completion.

An example demonstrating that experts can form deceptive overgeneralization, and that deceptive overgeneralization can lead to severe consequences, is the Crossair Flight 498 Crash. The official incident investigation report identified one human factor probable cause as follows: "when interpreting the attitude display instruments under stress, the commander resorted to a reaction pattern (heuristics) which he had learned earlier" [29, p.10].

As demonstrated in Figure 3, a Soviet attitude display indicates a left roll of the airplane with a counter-clockwise rotation. The appropriate response, detailed in Algorithm 1, is to stabilize the airplane by rotating it right. This rule acts as a cognitive shortcut that simplifies decision-making by minimizing the cognitive load needed to interpret the display. However, errors can arise if this shortcut is overgeneralized, omitting the condition that it should only apply to Soviet displays, leading to incorrect responses with other types of attitude displays.

Algorithm 1 Correct Production Rule for Interpreting (Soviet) Attitude Display to Stabilize an Airplane

```
if the goal is stabilize an airplane and attitude display
rotates counter-clockwise and it is a Soviet display
then
    rotate the airplane right
end if
```

For the first 20 years of his flying career, the commander received training that was "in theory comprehensive," exclusively at a flying school in the former Soviet Union [29, p.18]. However, upon transitioning to aircraft equipped with Western systems, no special differential training was provided to highlight the differences between Eastern and Western systems, nor did the commander undergo any unusual attitude training [29, p.19]. Therefore, the commander "had no opportunity to be trained in any other pattern of behavior" [29, p.96], meaning no opportunities to ever detect and correct the acquired deceptive overgeneralization. As the commander resorted to the overgeneralization in the scenario as illustrated in Figure 4, the commander kept rotating the airplane right (further) when the airplane was already rolling right, eventually resulting in a loss of control.

The acquisition of shortcuts can be modeled using the

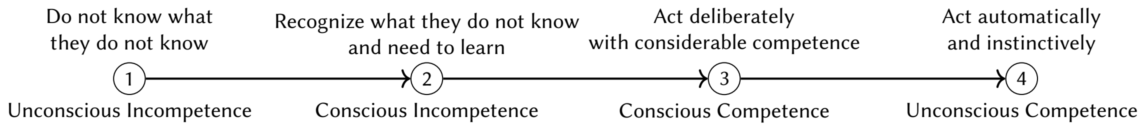
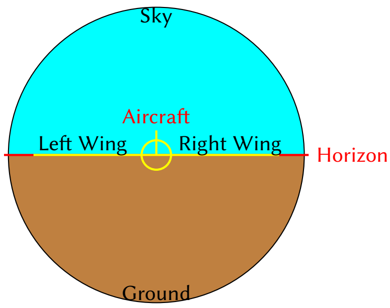
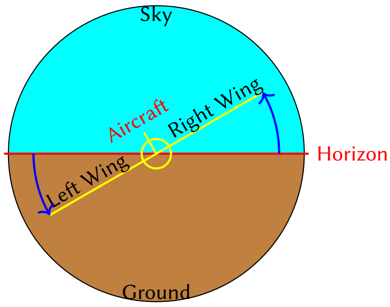


Figure 2: The Four Stages of Mastery. This model illustrates the progression from novice to expert, highlighting the development of competence and the shifting levels of consciousness.

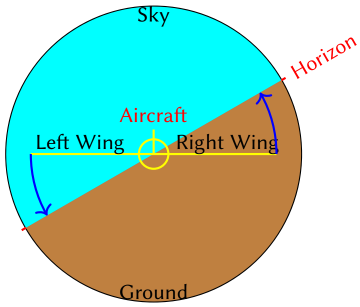


(a) Before counter-clockwise rotation



(b) After counter-clockwise rotation

Figure 3: A simplified depiction of a Soviet attitude display. The display reflects a "third-person view", where the horizon stays fixed, and the airplane's position is shown relative to the horizon. A counter-clockwise rotation (of the airplane relative to the horizon) indicates that the airplane is rolling **left**.



(a) After counter-clockwise rotation

Figure 4: A simplified depiction of a Western attitude display. The display reflects a "first-person view", where the airplane stays fixed and the horizon rotates relative to the airplane. A counter-clockwise rotation (of the horizon relative to the airplane) indicates that the airplane is rolling **right**.

process called *knowledge compilation* in the ACT-R theory, which serves to eliminate multiple production firings and the need for retrieval from declarative memory [4, p.169]. A primary compilation process, known as *composition*, is to takes sequences of productions that follow each other in solving a particular problem and collapses them into a single "macro-production" that has the effect of the sequence [2, p.235]. For example, Algorithm 1 could be compiled as shown in Algorithm 2. These production rules are intentionally represented in pseudo code, mimicking the implementation style of cognitive tutors developed with CTAT [24]. This representation serves to highlight several benefits of composition: fewer conditions and actions, fewer variables to track, and the elimination of redundant subgoals. These optimizations enhance the efficiency of the macro-production compared to the original series of separate productions [5, p.35].

However, it is possible that even experts who have mastered accurate basic production rules may develop inaccurate "macro-productions" during the process of building proficiency and automaticity if errors enter into the compilation process. Although composition increases overall efficiency by pruning redundant conditions and actions, these composed macroproductions tend to grow larger, particularly with an increase in the size of the condition sides [2, p.239]. With an increasingly more complex and composite condition side, it becomes more likely that some conditions will be overlooked, potentially leading to overgeneralization. While human compilation is gradual (in contrast to computer compilation), which may provide some protection against errors of omitting conditional tests from entering compilation, this protection is not infallible and can only reduce, but not eliminate, the possibility of condition omission [5, p.46].

Knowledge compilation in ACT-R theory suggests that new productions generated through knowledge compilation do not replace, but rather coexist with old ones [2, p.237]. A process known as conflict resolution then determines which productions to apply [2, p.132]. This raises

Algorithm 2 Knowledge Compilation for Interpreting Attitude Display to Stabilize an Airplane

Rule P1:
Condition: goal == stabilizeAirplane AND rollDirection == unknown
Action: subgoal = identifyRollDirection
Rule P2:
Condition: subgoal == identifyRollDirection AND displayRotation == counterClockwise AND displayType == Soviet
Action: rollDirection = left
Rule P3:
Condition: goal == stabilizeAirplane AND rollDirection != unknown
Action: subgoal = recoverAttitude
Rule P4:
Condition: subgoal == recoverAttitude AND rollDirection == left
Action: rotateAirplane(right)
Composed Rule P1&P2&P3&P4:
Condition: goal == stabilizeAirplane AND displayRotation == counterClockwise AND displayType == Soviet
Action: rotateAirplane(right)
Efficiency Gain:
2 subgoals, 4 conditions, 3 intermediate cognitive actions, and 2 variables get reduced by composition

the question of why the commander chose the overgeneralized shortcut over the basic alternative productions. The ACT-R strengthening mechanism might provide an explanation [2, p.250]. Production strength reflects the frequency of successful past applications [2, p.133]. Over the years, while flying Soviet aircraft, this shortcut—despite being overgeneralized—consistently led to correct actions within the context of Soviet attitude displays. This increased production strength may have made this shortcut the preferred choice during conflict resolution.

Another contributing factor to the commander's selection of the overgeneralization could be the medication effects, which potentially limited the commander's cognitive capacity [29, p.107]. The improved efficiency of the composed shortcut may have prompted the commander to favor the overgeneralized macro-production over a sequence of basic productions, especially under stress requiring immediate action, and possibly while multitasking. Such demanding and stressful scenarios are common, particularly in fields where individuals are considered experts and carry critical responsibilities. Moreover, situations involving limited cognitive capacity can occur to anyone. The ability to perform under conditions of stress, sleep deprivation, or fatigue is crucial, as is the capability to effectively manage simultaneous secondary tasks [30]. This indicates that overgeneralized shortcuts may be widespread, which highlights the importance of understanding their mechanisms through research.

The commander's extensive experience, amounting to over 8,000 hours [29, p.15], categorizes him within Stage 4 of the mastery model illustrated in Figure 2, where individuals are capable of acting automatically and instinctively. However, this incident starkly demonstrates that such automatic actions performed by experts, when based on deceptive overgeneralization, can lead to dire consequences. A similar case, that exemplifies the dangers of overgeneralization in aviation training, is the American Airlines Flight 587 crash, where *poorly-designed training led to deceptive overgeneralization, resulting in actions deemed correct during training but were inappropriate for actual conditions, ultimately leading to catastrophic outcomes.* Specifically, the American Airlines Advanced Aircraft Maneuvering Program included an excessive bank angle simulator exercise intended to prepare pilots for extreme wake turbulence. This equipped trainees with aggressive roll upset recovery techniques. Unfortunately, the scenario used in training was overly extreme and not representative of the actual aircraft type involved. This inappropriate training "enabled" the first officer to mistakenly apply these excessive techniques during a moderate wake turbulence encounter, leading to the inflight separation of the vertical stabilizer and culminating in a fatal plane nosedive [31]. It can be argued that had the pilot not been trained to perform such aggressive maneuvers, the disaster could have been entirely avoided.

In summary, acquiring a production rule that pairs correct actions with incorrect conditions is an undesirable learning outcome, which at best might later be rectified without severe repercussions, and at worst, could result in catastrophic outcomes.

3.4. Summary

This section presents the problem identification and examination on the phenomenon of deceptive overgeneralization through literature review and case studies, yielding several key characteristics of deceptive overgeneralization that underscore the need for further investigation:

1. Deceptive overgeneralization is prevalent across various domains.
2. Deceptive overgeneralization can be "sticky", difficult to detect and resistant to change.
3. In certain cases, deceptive overgeneralization can be worse learning outcomes than if the skill had not been learned at all.
4. Both novices and experts could be prone to deceptive overgeneralization.

Table 2
Summary of Methodologies for Each Research Question

Research Question	Methodology
RQ1: Formation	Experiments followed by Think-Aloud Studies; RCTs.
RQ2: Detection and Diagnosis	RCTs
RQ3: Remediation	RCTs
RQ4: Retrospective Discovery	EDM techniques using both synthetic and authentic datasets

4. Research Questions

My doctoral research aims to investigate the mechanisms of deceptive overgeneralization using the context of ITS and develop effective strategies for addressing deceptive overgeneralization. The proposed research questions are structured to methodically examine the formation, detection, remediation, and retrospective discovery of deceptive overgeneralization:

RQ1: Formation of Deceptive Overgeneralization. What types of production rules are most susceptible to deceptive overgeneralization? Under what conditions do ITS risk promoting deceptive overgeneralization?

RQ2: Detection and Diagnosis of Deceptive Overgeneralization. What features can be integrated into ITS to detect and diagnose deceptive overgeneralization?

RQ3: Remediation of Deceptive Overgeneralization. What instructional strategies are effective at correcting deceptive overgeneralization?

RQ4: Retrospective Discovery of Past Deceptive Overgeneralization. Can Educational Data Mining (EDM) techniques discover previously undetected deceptive overgeneralization from existing education datasets?

5. Methodology

This section has outlined the research methodologies corresponding to each of the research questions guiding my doctoral study. To rigorously investigate the phenomenon of deceptive overgeneralization, a diverse methodological approach will be employed. The methods range from experiments, think-aloud studies, and EDM techniques, as summarized in Table 2.

RQ1: Formation of Deceptive Overgeneralizations. The initial step in my research is to evaluate the hypothesized design flaw, as illustrated in Figure 1. This hypothesis suggests that when a series of practice activities only evaluate whether learners have performed the expected actions, such instructional designs may not adequately determine whether learners have internalized the correct rule or an overgeneralization.

My research strategy includes conducting experiments with ITS that adhere to best practices in ITS design, such as cognitive model development through Cognitive Task Analysis (CTA) [32], tailored hints and feedback, and task-loop adaptivity. However, these systems are not specifically designed to prevent deceptive overgeneralization. My experimental design draws inspiration from studies on the Einstellung effect, which describes how practice with a fixed method can bias individuals toward applying this method even when better alternatives exist [33]. In my experiments, learners will practice using ITS until they have achieved mastery as deemed by ITS. Subsequently, these learners will face tasks where the actions they have learned are no longer suitable. As my research contends that ITS may have limitations when it comes to accurately assessing true skill mastery, the research plan will incorporate qualitative data collected through think-aloud studies [34]. Specifically, "graduated novices"—learners who have completed training and are judged by the ITS to have mastered the content—will verbalize their understanding of the conditions during these sessions, in order to identify instances of deceptive overgeneralization. Next, to ascertain under what conditions ITS may inadvertently promote deceptive overgeneralization and to identify which features of instructional design are most susceptible to fostering these errors, my research plan includes conducting randomized controlled trials (RCTs) that compare different ITS interface designs and problem sequencing.

RQ2: Detection and Diagnosis of Deceptive Overgeneralization. To investigate features that can be integrated into ITS for effectively detecting and diagnosing deceptive overgeneralization, RCTs will be conducted to compare different ITS interface designs and problem sequencing.

Traditionally, ITS interfaces are designed to guide learners toward correct actions, potentially neglecting interface elements which represent potential incorrect actions that learners should avoid, as these elements do not belong to the prescribed solution pathway. Consequently, learners might attempt to perform incorrect actions but find themselves unable to do so, making those mistakes undetected, uncorrected, and unlogged. One hypothesized effective design is to provide practice opportunities where "lack of action" is the correct response. Although detecting non-actions poses more challenges than evaluating actions, we may consider ITS design that incorporates interface elements that learners should avoid interacting with, in order to make "lack of action" observable and test whether learners can appropriately refrain from actions when the conditions do not warrant them. This approach is similar to including distractor options in multiple-choice questions (MCQs), where learners must correctly identify and decide against choosing such options. Of course, the expertise reversal effect [11] suggests that such distractor interface elements should only be introduced when learners have reached a certain level of skill mastery, to ensure that cognitive workload remains manageable.

RQ3: Remediation of Deceptive Overgeneralization. Similar to RQ2, RCTs that compare different ITS interface designs and problem sequencing will be conducted. One instructional design hypothesized to be effective involves providing side-by-side comparisons between scenarios that do and do not warrant certain actions. This approach requires learners to identify differences in problem features, facilitating a deeper understanding of when specific actions are appropriate.

Incorporating both RQ2 and RQ3, the problem sequencing design pattern illustrated in Algorithm 3 is hypothesized to aid both in initial induction and subsequent refinement, and can detect, diagnose, and remedy deceptive overgeneralization. The `checkSAI()` function, as in CTAT, represents the automated evaluation by ITS that compare learner actions with reference ones [24].

Algorithm 3 Problem Sequencing Design Hypothesized to Aid in Initial Induction and Subsequent Refinement

```
Target Knowledge Component (KC):
if A AND B then
    <action>
end if
Potential Overgeneralization:
if A then
    <action>
end if
Problem Type 1: Designed for Induction
if A AND B then
    checkSAI(<action>)
end if
Problem Type 2: Designed for Refinement
Problem Subtype 2.1: Unsuitable Context
if A AND NOT B then
    checkSAI(NO <action>)
end if
Problem Subtype 2.2: Insufficient Information
if A AND Missing Info about B then
    checkSAI("Not Enough Info")
end if
```

RQ4: Retrospective Discovery of Past Deceptive Overgeneralization. In addition to designing and conducting experiments specifically for investigating deceptive overgeneralization, my research could better contribute to the TEL community if there is evidence that the research findings can also generate actionable insights using existing datasets. Therefore, the last research question focuses on retrospective analysis to discover past deceptive overgeneralizations, using learning datasets already collected through standard procedures. My research plans to employ *learning curve* analysis facilitated by DataShop [35], which graphically represents changes in learner performance, visualizing any improvement or stagnation as learners engage in repeated practice opportunities [36]. ITS systems developed with CTAT, which typically store learning logs in DataShop, which are ready candidates for retrospective analysis.

To effectively visualize and demonstrate learning curves that may indicate overgeneralization, I will start with synthetic data. Synthetic data, artificially generated by computer algorithms and not derived from real-world events, mimics authentic datasets. The ethical generation and application of synthetic data is a widely accepted practice in learning sciences, particularly within the realm of Educational Data Mining (EDM), as evidenced by its use in numerous EDM research studies [37, 38, 39, 40, 41]. Synthetic data addresses the complexities of authentic learner data, aiding in the validation of models for skill mastery assessment, and can faithfully reflect reality when properly modeled [41].

To examine how deceptive overgeneralization affects learning trajectories, BKT was used to simulate performance with problem sequencing illustrated in Algorithm 3 with the following parameters: $p_{initial} = 0.5$, $p_{transition} = 0.2$, $p_{slip} = 0.1$, and $p_{guess} = 0.2$. The learning process is modeled with a single KC with three possible states: *Unlearned*, *Overgeneralized*, and *Learned*. This approach adheres to the BKT framework by treating the learning progression as a transition between states. As learners in the *Unlearned* state receive repeated practice opportunities, they may either remain in the *Unlearned* state, transition to an *Overgeneralized* state, or move directly to the *Learned* state. Learners in the *Overgeneralized* state can only possibly progress to the *Learned* state through problems designed for refinement. Another core assumption made in the simulation is the probability of correct responses based on knowledge state and problem phase, as illustrated in Table 3. Problems designed for induction can be correctly answered (unless a slip occurs) using either the correct generalization or an overgeneralization. For the problems designed for refinement, learners who either remain in the *Unlearned* state or who have adopted the overgeneralized rule are expected to answer incorrectly most of the time. However, rather than guessing like those in the *Unlearned* state, learners in the *Overgeneralized* state will answer incorrectly unless a slip occurs, which reflects how learners with deceptive overgeneralization will “confidently” make mistakes when the conditions do not actually warrant the actions.

Table 3
Probability of Correct Responses Based on Knowledge State and Problem Phase

State	Induction Phase (Same as Original BKT)	Refinement Phase (Reverse Slip Model)
Unlearned	P_GUESS	P_GUESS
Overgeneralized	1 - P_SLIP	P_SLIP
Learned	1 - P_SLIP	1 - P_SLIP

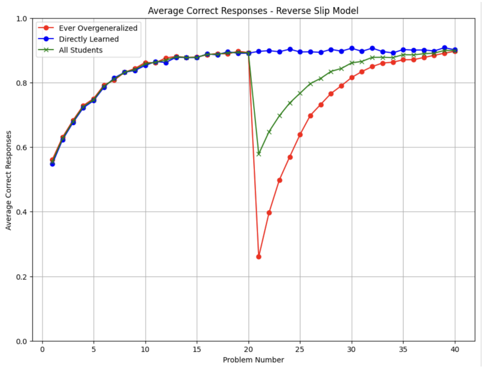


Figure 5: Simulated Performance Trends

Figure 5 visualizes the simulated performance trends of learners with the above assumptions. First, during the induction phase, the performance is not distinguishable between the "Ever Overgeneralized" group and the "Directly Learned" group. Second, the "Ever Overgeneralized" group (red line), with learners who have ever acquired the *Overgeneralized* state, notably exhibits a significant and sudden performance drop when transitioning to the refinement phase, which corresponds to the problems designed to detect overgeneralization. This drop starkly contrasts with the stable performance growth of the "Directly Learned" group (blue line) with learners who directly transited from *Unlearned* to *Learned* state. The performance recovery of the "Ever Overgeneralized" group after the drop demonstrates the remediation of overgeneralization.

My future research plan is to transition from synthetic to authentic datasets by collaborating with other researchers to perform retrospective analysis on existing datasets.

6. Contribution to TEL

In my doctoral research, I plan to extend the ACT-R cognitive architecture to tackle deceptive overgeneralization. My research seeks to refine the adaptivity of ITS and enable more accurate assessments of true skill mastery. This work contributes to Technology-Enhanced Learning (TEL) by enhancing the precision of automated assessments and supporting more reliable adaptive learning experiences.

References

[1] V. Aleven, E. A. McLaughlin, R. A. Glenn, K. R. Koedinger, Instruction based on adaptive learning technologies, Handbook of research on learning and instruction 2 (2016) 522–560.

[2] J. R. Anderson, The Architecture of Cognition, Harvard University Press, USA, 1983.

[3] J. R. Anderson, Act: A simple theory of complex cognition., American psychologist 51 (1996) 355.

[4] J. R. Anderson, Automaticity and the ACT theory, The American journal of psychology (1992) 165–180.

[5] J. R. Anderson, Acquisition of cognitive skill., Psychological review 89 (1982) 369.

[6] V. Aleven, B. M. McLaren, J. Sewall, M. Van Velsen, O. Popescu, S. Demi, M. Ringenberg, K. R. Koedinger, Example-tracing tutors: Intelligent tutor development for non-programmers, International Journal of Artificial Intelligence in Education 26 (2016) 224–269.

[7] K. VanLehn, The behavior of tutoring systems, International journal of artificial intelligence in education 16 (2006) 227–265.

[8] B. P. Woolf, Building intelligent interactive tutors: Student-centered strategies for revolutionizing e-learning, Morgan Kaufmann, 2010.

[9] L. S. Vygotsky, M. Cole, Mind in society: Development of higher psychological processes, Harvard university press, 1978.

[10] A. Collins, J. S. Brown, A. Holum, et al., Cognitive apprenticeship: Making thinking visible, American educator 15 (1991) 6–11.

[11] S. Kalyuga, The expertise reversal effect, in: Managing cognitive load in adaptive multimedia learning, IGI Global, 2009, pp. 58–80.

[12] K. R. Koedinger, V. Aleven, Exploring the assistance dilemma in experiments with cognitive tutors, Educational Psychology Review 19 (2007) 239–264.

[13] N. R. Council, How People Learn: Brain, Mind, Experience, and School: Expanded Edition, The National Academies Press, Washington, DC, 2000. doi:10.17226/9853.

[14] V. J. Shute, Focus on formative feedback, Review of educational research 78 (2008) 153–189.
[15] S. A. Ambrose, M. W. Bridges, M. DiPietro, M. C. Lovett, M. K. Norman, How learning works: Seven research-based principles for smart teaching, John Wiley & Sons, 2010.
[16] K. R. Koedinger, A. Corbett, et al., Cognitive tutors: Technology bringing learning sciences to the classroom, na, 2006.
[17] A. T. Corbett, J. R. Anderson, Knowledge tracing: Modeling the acquisition of procedural knowledge, User modeling and user-adapted interaction 4 (1994) 253–278.
[18] J. R. Anderson, C. D. Schunn, Implications of the act-r learning theory: No magic bullets, in: Advances in instructional Psychology, Volume 5, Routledge, 2013, pp. 1–33.
[19] K. R. Koedinger, A. T. Corbett, C. Perfetti, The knowledge-learning-instruction framework: Bridging the science-practice chasm to enhance robust student learning, Cognitive science 36 (2012) 757–798.
[20] V. A. Aleven, K. R. Koedinger, An effective metacognitive strategy: Learning by doing and explaining with a computer-based cognitive tutor, Cognitive science 26 (2002) 147–179.
[21] H. Zhao, K. Koedinger, J. Kowalski, Knowledge tracing and cue contrast: Second language english grammar instruction, in: Proceedings of the Annual Meeting of the Cognitive Science Society, volume 35, 2013.
[22] N. M. Chang, Learning to discriminate and generalize through problem comparisons, Ph.D. thesis, Carnegie Mellon University, 2006.
[23] LearnLab, Feature validity, 2011. URL: https://learnlab.org/wiki/index.php?title=Feature_validity, [Online; accessed 29-May-2024].
[24] V. Aleven, B. McLaren, J. Sewall, K. R. Koedinger, Example-tracing tutors: A new paradigm for intelligent tutoring systems (2009).
[25] V. Aleven, B. M. McLaren, J. Sewall, K. R. Koedinger, The cognitive tutor authoring tools (CTAT): Preliminary evaluation of efficiency gains, in: Intelligent Tutoring Systems: 8th International Conference, ITS 2006, Jhongli, Taiwan, June 26-30, 2006. Proceedings 8, Springer, 2006, pp. 61–70.
[26] K. R. Koedinger, J. R. Anderson, W. H. Hadley, M. A. Mark, Intelligent tutoring goes to school in the big city, International Journal of Artificial Intelligence in Education 8 (1997) 30–43.
[27] M. T. Chi, P. J. Feltovich, R. Glaser, Categorization and representation of physics problems by experts and novices, Cognitive science 5 (1981) 121–152.
[28] B. H. Ross, Remindings and their effects in learning a cognitive skill, Cognitive psychology 16 (1984) 371–416.
[29] A. A. I. Bureau, Final report of the aircraft accident investigation bureau on the accident to the saab 340b aircraft, registration hb-akk of crossair flight crx 498 on 10 january 2000 near nassenwil/zh, 2002.
[30] R. A. Schmidt, R. A. Bjork, New conceptualizations of practice: Common principles in three paradigms suggest new concepts for training, Psychological science 3 (1992) 207–218.
[31] S. Board, In-flight separation of vertical stabilizer american airlines flight 587 airbus industrie a300-605r, n14053 belle harbor, new york november 12, 2001, National Transportation Safety Board 490 (2001).
[32] J. M. Schraagen, S. F. Chipman, V. L. Shalin, Cognitive task analysis, Psychology Press, 2000.
[33] A. S. Luchins, Mechanization in problem solving: The effect of einstellung., Psychological monographs 54 (1942) i.
[34] K. A. Ericsson, H. A. Simon, How to study thinking in everyday life: Contrasting think-aloud protocols with descriptions and explanations of thinking, Mind, Culture, and Activity 5 (1998) 178–186.
[35] K. R. Koedinger, R. S. Baker, K. Cunningham, A. Skogsholm, B. Leber, J. Stamper, A data repository for the edm community: The pslc datashop, Handbook of educational data mining 43 (2010) 43–56.
[36] K. R. Koedinger, R. S. Baker, K. Cunningham, A. Skogsholm, B. Leber, J. Stamper, A data repository for the edm community: The pslc datashop, Handbook of educational data mining 43 (2010) 43–56.
[37] M. M. Rahman, Y. Watanobe, T. Matsumoto, R. U. Kiran, K. Nakamura, Educational data mining to support programming learning using problem-solving data, IEEE Access 10 (2022) 26186–26202.
[38] N. Ndou, R. Ajoodha, A. Jadhav, Educational data-mining to determine student success at higher education institutions, in: 2020 2nd International Multidisciplinary Information Technology and Engineering Conference (IMITEC), IEEE, 2020, pp. 1–8.
[39] J. M. Patil, S. R. Gupta, Extracting knowledge in large synthetic datasets using educational data mining and machine learning models, in: Soft Computing for Intelligent Systems: Proceedings of ICSCIS 2020, Springer, 2021, pp. 167–175.
[40] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. J. Guibas, J. Sohl-Dickstein, Deep knowledge tracing, Advances in neural information processing systems 28 (2015).
[41] M. C. Desmarais, I. Pelczer, On the faithfulness of simulated student performance data, in: Educational Data Mining 2010, 2010.