Behavior Research Methods
2008, 40 (3), 673-693
doi: 10.3758/BRM.40.3.673

Train-to-Code: An adaptive expert system for training systematic observation and coding skills

Jessica M. Ray
University of Central Florida, Orlando, Florida

and

Roger D. Ray
Rollins College, Winter Park, Florida

Problems in training behavioral observers to a high degree of interindividual accuracy and intraindividual stability are fundamental concerns in descriptive research, as well as in provisions of behavioral intervention services. This article presents design characteristics of and results from three formative evaluations of an adaptive computerized expert system that shapes observation and recording skills and maximizes both individual coding accuracy and stability. The system, called *Train-to-Code*, allows instructors or trainers to import their own video source files and to code those videos using any appropriate descriptive behavioral-coding scheme. This generates customized expert reference data that automate subsequent training on the basis of an operant response-shaping instructional design model. Successful training relies on transitions through alternative levels of prompting and feedback designed to optimize ongoing performance until stable expert-equivalent levels of interobserver accuracy are maintained without prompting or feedback.

Many publications on systematic behavioral observation methods (see Bakeman & Gottman, 1997; Bass, 1987; Johnston & Pennypacker, 1993; Lehner, 1998; Martin & Bateson, 1993; Sackett, 1978; Sharpe & Koperwas, 2003) have emphasized problematic issues in attaining maximal interobserver accuracy and intraobserver recording consistency. Although theoretical and methodological discussions often have focused on observer calibration and reliability decay (see Bakeman & Gottman, 1997), we have found little empirical research on intraindividual variability in accuracy (see Bass, 1987; Patterson & Moore, 1979; Taplin & Reid, 1973). In fact, we have found no publication that has reported stability in coding accuracy or reliability using momentary point-by-point time series measures of accuracy *within* coding sessions and only one (Bass, 1987) that has systematically tracked accuracy and variability as a time series *across* successive sessions.

In part, this paucity of research is a product of former technological limitations for measuring moment-by-moment variations in observer accuracy. Nevertheless, implications of accuracy variation extend beyond theoretical or methodological concern. Accuracy calibration, decay, and stability in detecting ongoing behavior states and state changes are also fundamental to multistaff-delivered behavioral interventions. For example, consistencies in client contingency management and staff performance evaluations within residential treatment facilities have highly practical implications and typically involve substantial expert-trainer time and expense. This article introduces a new technology that addresses such training and assessment issues based on an expert-referenced intelligent training system (ITS). This system is designed to train and calibrate individual observational and coding behaviors using automated running averages of entry-by-entry accuracy measures.

VIDEO AND COMPUTERS IN OBSERVER TRAINING

By the early 1980s, psychological researchers began to recognize the value of introducing video into observational training as a means of addressing both effectiveness and efficiency. For example, Van Der Molen, Kerkhof, and Jong (1983) suggested that although field experience was still an important part of the training process, video-based training for complex behavioral coding was more efficient than field-based training. By the late 1980s, the continuing need for improved methods led Bass (1987) to explore the effectiveness of computer-assisted interventions in video-based observer training. Bass's study is noteworthy for several innovations beyond our prior note of his across-sessions comparison of accuracy. He presents one of the rare studies we have found that has systematically explored alternative levels of data source complexity, taxonomic demand characteristics, and alternative levels of support given during training.

R. D. Ray, rdray@rollins.edu

 Copyright 2008 Psychonomic Society, Inc.

A few additional computer-based observation-related programs have been developed over the subsequent decades (e.g., COR by Blasko, Kazmerski, Corty, & Kallgren, 1998; OBSERVE by Dickins et al., 2000; and CORv2 by Blasko, Kazmerski, & Torgerson, 2004). However, these programs tend to emphasize the teaching of conceptual issues embedded in observational methods and procedures or to offer assisted data recording and automated data analysis, rather than actually training observational discrimination and coding skills per se. Thus, although existing programs represent various alternative instructional models for incorporating video and computer technologies into observational instruction, most remain static or passive in efforts to develop coding skills and offer only slight improvement over more traditional textbook instructions. Put simply, these systems focus more on informing than on skill training. Nevertheless, advances in *informational tutoring* system designs offer significant guidance to inform the design for computer-based skill training.

Modern artificially intelligent computer-based tutoring systems—another form of ITS—typically stress text-based presentations of stimuli and question answering, rather than focusing on perceptual–motor skill performance (see Graesser, Jackson, & McDaniel, 2007; Graesser et al., 2004; R. D. Ray, 2004; R. D. Ray, Gogoberidze, & Begiashvili, 1995). At their best, such programs incorporate intelligent and goal-adaptive expert models to offer services that simulate the dynamic and changing nuances of one-on-one human instruction (R. D. Ray, 2004). Such ITSs are based on an instructional model that incorporates both (1) *knowledge generation* engines that model the developing student skill or knowledge repertory and (2) a set of alternative training goals or objectives that dynamically adjust to evolving changes in student performances (R. D. Ray & Belden, 2007). Such systems compare student progress with an expert reference model for purposes of individualized and informed content and prompt presentation, tutoring/training, feedback, and objectives implementation.

As has been noted, this article introduces a new observer training system, called *Train-to-Code* (TTC), that incorporates an adaptive expert systems model for computerized training. More specifically, TTC trains observers via an operant response-shaping instructional model (R. D. Ray, 1995; R. D. Ray et al., 1995) that guides training by incorporating a gradation of *successive approximations*, or behavioral steps, to bridge the gap between novice skills and expert skills via a succession of escalating and targeted skill and support stages. As we apply it, shaping is a personally focused procedure used to facilitate a student's progression through successive skill development stages by supplying various degrees of stimulus prompting and accuracy-driven feedback as reinforcement. We begin with high-density prompts and feedback that systematically fade to lower densities until no prompting and no feedback are given at all. Readers unfamiliar with the operant shaping process and its emphasis on the differential reinforcement of successive approximations to a goal response class will find a substantial discussion of the process in Catania (1998) and a historic discussion of Skinner's discovery of the process in Peterson (2004).

TRAIN-TO-CODE Design Features and Formative Evaluation Strategies

TTC offers two alternative user interfaces, or *modes of use*: the instructor/trainer mode and the student mode. Both of these modes incorporate feature sets that are currently evolving on the basis of formative and evaluative research (Layng, Stikeleather, & Twyman, 2006), as exemplified by the research presently reported. In this section, we will describe features that were the foundation for our initial experiments. We subsequently will detail alterations and additions made to those features, after reporting the specific experimental results that led to respective modifications.

Instructor Mode

The instructor mode of TTC allows trainers or instructors to develop any number of customized expert training *setups*, using their own digital video file(s) and their own behavioral taxonomy or coding scheme. By applying any given taxonomy to code one or more videos, and by saving this coding as an *expert reference* file, instructors generate a customized personal training system. The TTC system may eventually allow an instructor to select and teach any 1 of 14 alternative recording procedures (J. M. Ray, Gillins, & Ray, 2006), while also teaching accurate and stable use of any given (single-domain) coding scheme. One of two currently implemented recording procedures, *behavior frequency count* (BFC), creates a simple frequency-of-occurrence count reflecting the total number of instances of each behavior category occurring throughout an entire observation session. The second implemented recording procedure, *continuous coding* (CC), requires use of an exhaustive category taxonomy and retains both the sequence and the corresponding temporal onset/offset of each successive behavioral event. This is the required procedure for generating all expert reference coding, thus supplying *state* information for all successive frames of corresponding video source files.

TTC's design affirms that, to be a training system rather than an informational instruction system, a system must do more than present instructional text. And to be a truly adaptive training system, an application must offer varying prompting assistance and discriminative response feedback as adaptive services tailored to a student's changing skill levels. This is achieved by implementing various levels of adjustable video play rates, event discrimination prompting, and coding response feedback, with these features being gradually faded as coding accuracy increases—thus shaping a student's coding performance to accuracies equivalent to those of an expert observer/recorder. Parameters controlling adaptive services in TTC have default settings but are also accessible by instructors via a training management panel for parameter adjustment or empirical evaluation (see the upper-right corner of Figure 1). The instructor mode also allows for tracking a student's progress through time series graphs of observer

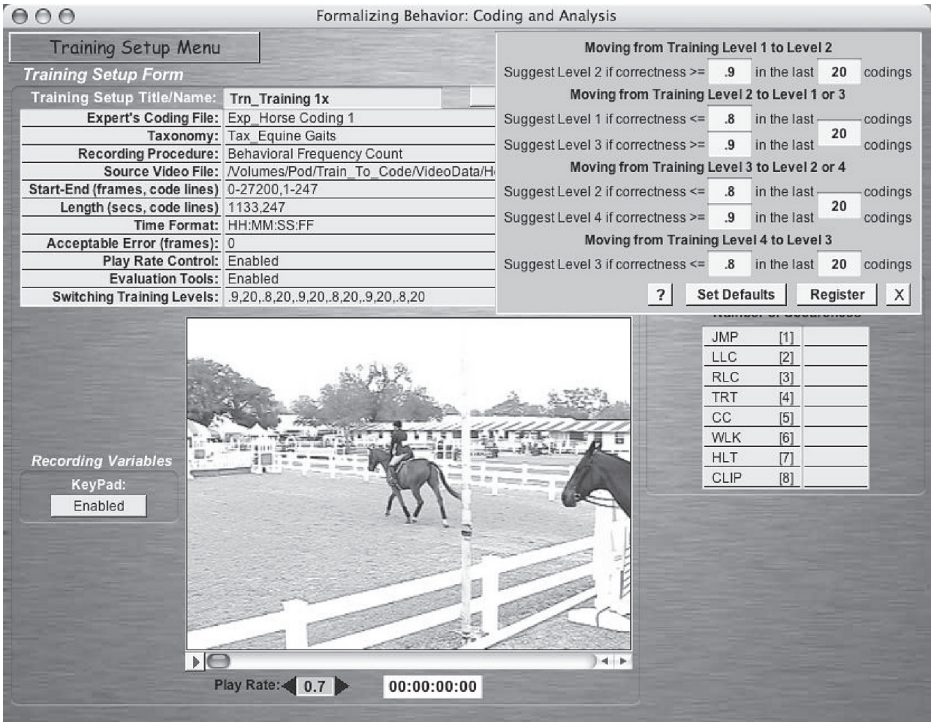

Figure 1. Instructor's panel for training setup and adaptivity parameters.

accuracy measures, as Van Der Molen et al. (1983) have defined this term.

Student Mode

The student mode in TTC consists of three alternative student services. These three services include (1) a *foundations* section that provides three forms of video-illustrated definitional material designed to teach the terms and definitions that make up a taxonomy, (2) a *training* section that includes four levels of adaptive training targeting behavioral coding skills, and (3) a *certification* section for testing and unassisted performance evaluation.

Foundations. The foundations section of TTC is subdivided into three levels. The first level offers operational definitions for each behavior in a given taxonomy. The student learns these definitions through a mixed media presentation format. When the student clicks on a behavior category in the listing of taxonomic categories (or enters the corresponding keypad number, as would be done during a coding session), the student is presented a textual definition of the behavior, as well as one of a series of available video examples illustrating variations of that behavior.

The second level of foundations allows the student to manually click through a complete coding session, behavior by behavior, in the actual sequence coded by the expert. This allows the student to view behaviors in a controlled but natural sequence and to begin to discriminate important features defining each behavioral transition. The third level (see Figure 2) presents a similar opportunity for observation, but the manual control over each successive sequence is now replaced by an automated pause between behaviors. This level is intended to more closely approximate real-time transitions between behaviors, while also showing the behavioral term (code) that describes each component of that transition.

Training. In TTC, each student coding response is immediately compared with the expert model on the basis of the reference file described earlier. Each response is logged and marked as correct or incorrect (the *type* of error, based on several alternative possibilities, is recorded as well). Student–expert response comparisons are continually used to determine appropriate stimulus prompts and coding feedback, as well as point-by-point accuracy measurements and adaptive instructional services. These services define the adaptive training section, which consists of four successive approximation levels of alternative services described below. Relying upon our operant response-shaping instructional model discussed previously, visual timing prompts and error-correcting feedback messages are gradually faded as the student advances through the four levels. When performance suggests the need, the student may also be prompted to return to a lower level for more supportive training. The following sections detail the training methods used with the BFC recording procedure defined earlier. A similar program of adaptivity is used for alternative recording procedures, but specific prompting and feedback features are appropriately modified to be procedure-specific.

Training Level 1. The first level of training assumes only prior foundations exposure and is designed to provide maximum prompting with respect to behavioral timing and feedback. At Training Level 1, a student is prompted to observe each new behavioral occurrence by the presentation of a green frame surrounding the video

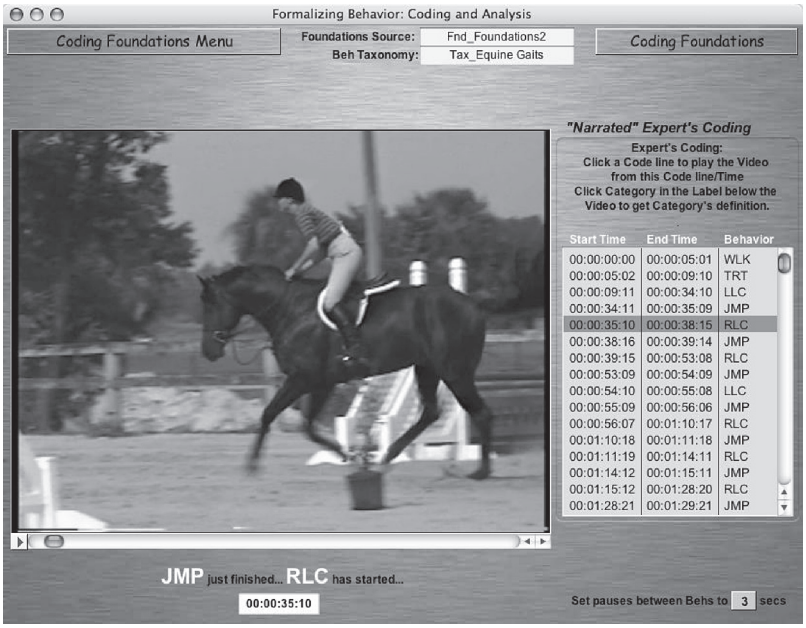

Figure 2. Foundations mode: Event labeling for a behavioral transition in Foundations 3.

playback area. This green frame continues for the first half of the behavior's total duration. If the student codes the behavior correctly during this prompt, feedback is given to indicate that the code entered was correct and timely (see Figure 3). This correct coding entry also turns off the framing prompt for the remainder of that behavior's duration. The green framing prompt reappears on the first frame of the next new behavior in continuous video play. If any behavior is not coded during the green "new behavior has started" prompt, the framing of the video window becomes yellow for the second half of the behavior's duration. If a code has not been entered by the final 2 sec of the behavior, the prompt frame changes from yellow to red—indicating the imminent end of the behavior.

If an incorrect code is entered at any time during the behavior, a feedback message appears indicating that the current behavior is not the code entered (e.g., if, say, the student is coding horse gaits and other behaviors and, in the coding

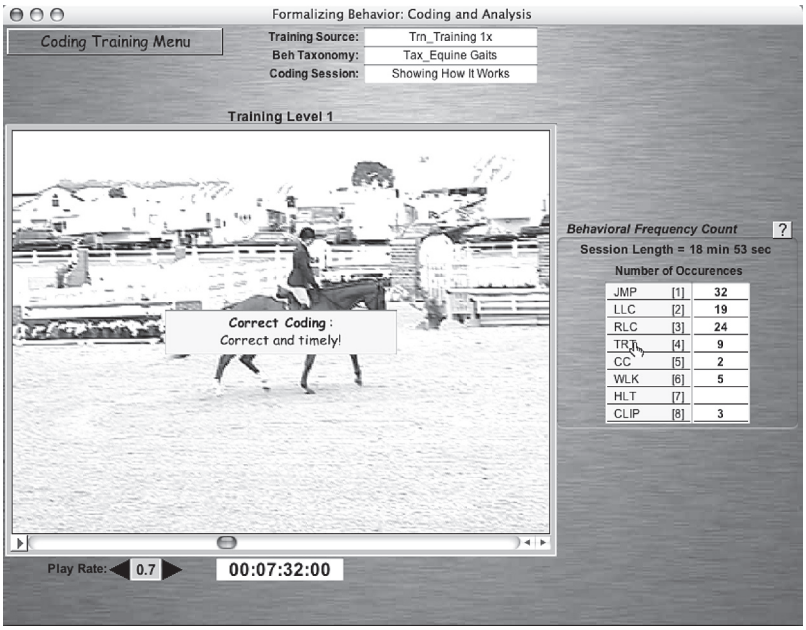

Figure 3. Student mode: Prompting and feedback at Training Level 1.

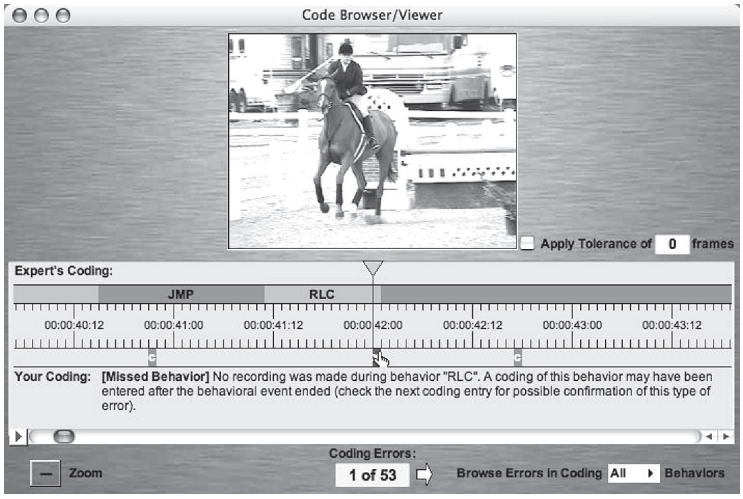

Figure 4. Code browser/viewer illustrates both expert and student codings for corresponding video frames on a time line. Student coding errors are marked with an "X," and roll-overs bring up explanations of the type of error made.

process, enters "RLC" [for "Right-Lead Canter"] during the behavior "Jump," a feedback text message immediately appears saying that "the current behavior is not RLC"). The video then replays the mislabeled behavior, thus allowing/requiring that specific behavior to be recoded. If a student fails to code a behavior altogether at Training Level 1, a feedback message appears indicating that the behavior was missed. The missed behavior event is then replayed from its beginning for correct coding. When a student's coding accuracy is at, or above, the instructor-set criteria for correctly coding at the current level, a message appears suggesting that the student choose to move to the next higher adaptive level—Level 2, in this case.

Training Level 2. At Training Level 2, the student's experience is similar to that at Training Level 1, but this level also incorporates some subtle and important differences. Some temporal prompting and feedback features have been faded at Training Level 2, as would likely happen during an in vivo training session. Prompts are now used only to indicate the beginning and end of a behavior, thus leaving the majority of the behavioral event unprompted. A green frame appears around the video for approximately 2 sec at the beginning of the behavior, and the red framing again signals the remaining 2 sec at the end of the behavior. There is no yellow framing shown at any time at this level. If a correct code is entered during a behavioral event, the video simply continues with neither feedback nor further prompting until the beginning of the next behavior. If an incorrect code is entered, a feedback message similar to that at Training Level 1 appears. If a behavior ends without having been coded during Training Level 2, a feedback message is presented informing the student that a behavior has just been missed, but the student is not returned to recode the behavior. Instead, the feedback message presents the correct coding.

Training Level 3. Feedback messages and prompts are further faded when a student moves to Training Level 3. At this level, all temporal framing prompts are faded, and no feedback is given for correct or incorrect code entries. Thus, the only "training" at this level is through presentations of a message when a behavioral event ends without coding (i.e., behavior was *missed*). Much like the feedback for missed behaviors in Training Level 2, if a student fails to code a behavior in Training Level 3, a feedback message is presented indicating the specific behavior just missed. When the feedback message disappears, play resumes, beginning with the next behavior.

Training Level 4. The final level of Training is Level 4, also called *self assess*. At this level, the student is assumed to be an expert coder. Thus, neither prompting nor feedback is given during coding. This training level simulates both an in vivo coding session and the conditions of mastery certification testing in TTC.

Auxiliary Feedback

In addition to the varying types of real-time feedback incorporated into the training levels described above, delayed-access auxiliary feedback is accessible during all levels of training by opening a pop-up window presenting a *code browser/viewer* (see Figure 4). In this viewer, coding entries are presented on a highly detailed frame-by-frame time line of the video. Above this time line, the expert's coding of the ongoing video is illustrated, and under the time line, the student's coding entries appear. A reference replay of the actual video is also provided within this tool to illustrate the corresponding footage for each coding entry being reviewed. The code viewer also allows the student or instructor to see each and every coding entry made, including the special marking of all errors, across a scrollable time line of the coded video segments. In addition, the timing of the coding input with respect to the total duration of each behavior is illustrated, thus allowing inspections of correct, but delayed, codings, as well as error codings.

The code viewer also allows a student or instructor to click directly through only the errors in a given coding

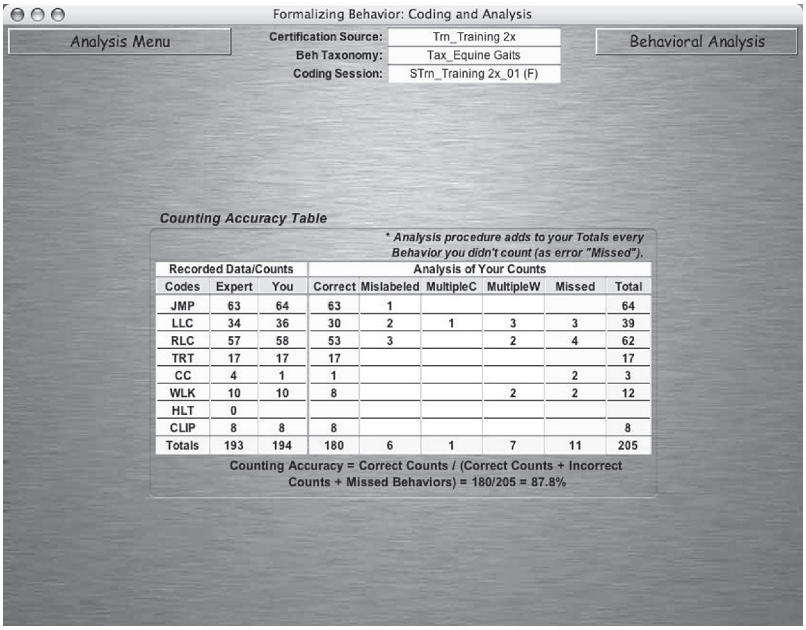

Figure 5. Coding accuracy table illustrating all coding inputs for expert and student, with frequency and types of errors listed for each behavioral category.

session, thereby giving quick and direct access to detailed information as to which codings were in error and why any given coding entry was incorrect. As has been noted, users may also view the corresponding video representation of each behavior while comparing student coding with the correct coding of the expert in frame-by-frame playback comparisons. It is expected that this feature will be especially useful in targeting timing errors, as well as clarifying repeatedly miscoded behaviors.

When using the BFC recording procedure, other feedback is available upon completion of an entire coding session. One form of feedback is a counting accuracy table. As is illustrated in Figure 5, this table reports the total student counts for each behavior category, in comparison with the expert's counts. This table also details the student's total counts broken down by categories that include correct, mislabeled, missed, multiple coding/wrong labeling, and multiple coding/correct labeling. Also available is a more traditional confusion matrix comparing expert versus student codings for each event (see Bakeman & Gottman, 1997). Accompanying the matrix is a panel of coding accuracy statistics based on Cohen's (1960) kappa and simple percentage-of-agreement scores. Alternative recording procedures offer similar procedure-specific feedback features. For example, CC procedures incorporate only the traditional confusion matrix with its appropriate statistics, along with access to the code viewer for detailed error evaluation. The counting accuracy table does not apply to this recording procedure (it is redundant to the confusion matrix in this case) and thus is not available.

Formative Evaluation Research Strategies

Layng et al. (2006) suggested that, when used properly as part of the design process, a formative research approach to building computerized instructional systems guarantees significant evaluative results once the product is completed. Inherently, such research creates a feedback loop between design theory and the working model. If probe data suggest that one or more components are not effective, the product design is modified until desired results are achieved, thus iteratively informing design decisions (Andronis, Twyman, & Stikeleather, 2004). As was noted earlier, it is under this formative research model that the TTC system is being developed and its efficacy investigated. We present three such experimental probes to illustrate (1) the strategy, (2) the success and failures of specific designs, and (3) the current direction our work is taking.

EXPERIMENTS 1 AND 2

Method

Participants. Three female undergraduate psychology majors at Rollins College participated in a multisession and multistage single-participant research design. For each session, a single participant was located at a computer station in one room while the experimenter monitored progress from another room via a one-way observational window.

Apparatus and Materials. All the participants coded via a numeric keypad built into a keyboard on a lower keyboard drawer that was separated from the desktop computer. Our initial research probes of TTC were designed with rudimentary sports-judging skills—specifically, equine hunter–jumper fundamentals. We thus generated a mutually exclusive and exhaustive taxonomy based on fundamental horse gaits and other competition-related behaviors, such as jumping and halting (see Table 1 for the taxonomy and definitions used). A corpus of home videos depicting various horses and riders was edited to include multiple competitive show *rounds*, with each round (defined by a separate entry/exit to/from a low-fenced show area via a gate) being separated by a 5-sec *black screen* video segment. Students were required to code each black screen segment as "CLIP" (see Table 1).

Table 1
Equine Behavior Taxonomy

Behavior	Keypad Number	Operational Definition
JMP (jump)	1	The jump begins when both forelegs leave the ground over an obstacle. This is followed by both hind hooves leaving the ground to clear the obstacle. The behavior ends when the trailing hind leg touches the ground on the landing side of the obstacle.
LLC (left-lead canter)	2	The canter is a three-beat gait. When on the left lead, the horse's leg movements are characterized by the right hind leg moving forward first, followed by the left hind and right front moving forward and landing together in time, and finally the left front moving forward and landing in front of all other legs.
RLC (right-lead canter)	3	The canter is a three-beat gait. When on the right lead, the horse's leg movements are characterized by the left hind leg moving forward first, followed by the right hind and left front moving forward and landing together in time, and finally the right front moving forward and landing in front of all other legs.
TRT (trot)	4	The trot is a two-beat gait in which the legs move in diagonal pairs. Thus, the right hind and left front move forward together in time, and the left hind and right front move together in time, creating diagonal movements.
CC (cross-canter)	5	The cross-canter is a three-beat gait in which the horse is cantering on one lead with his front legs and the opposite lead with the hind legs. Thus, leg movements might follow a pattern of left hind landing, right hind and right front, and finally left front.
WLK (walk)	6	The walk is a four-beat gait with the legs moving forward in lateral movements, such as right hind followed by right front, left hind, and finally the left front.
HLT (halt)	7	The halt is defined as the horse standing in one place with no forward, backward, or side movements more than one step in any direction.
CLIP	8	Clip indicates a black or gray screen not showing video. This is not a behavior but must be coded.

Procedure. BFC coding procedures required that a student recognize and code the occurrence of each type of behavioral/categorical event only while it was still in progress. All the sessions relied upon this BFC procedure, which we introduced with instructions that emphasized the need to code as quickly as possible following the onset of each behavioral event. Unlike field-based BFC coding based on real-time rates of occurrence, for training purposes (mainly because some categories, such as jump, are extremely brief in duration), all coding sessions presented video with play rates set to .7 of the real-time 30-fps normal video play rate. Upon each keypad input, TTC immediately assessed coding accuracy and calculated interobserver agreements from unambiguous synchronized comparisons between expert and student codings. Thus, a running average of an adjustable number of input events was used in TTC to calculate accuracy ratios that reflected percentage of agreement (correct inputs divided by correct-plus-errors). Critical values alerted a participant as to the appropriate level of support the participant was required to accept when the program suggested a change in levels. Normally, it is a user option to change or remain on current levels, but accepting recommended change was required in all the experiments.

The first experimental stage (baseline) involved giving a participant a printed handout with a category-by-category listing of the coding taxonomy, along with corresponding definitions for each category. We asked the participants to study the handout and, subsequently, to refer to it while coding a series of six successive rounds. In sessions following baseline coding assessment, we asked the participants to review all three levels of foundations training for approximately 20 min and, subsequently, to code a new set of five additional rounds (postfoundations).

The subsequent condition (Experiment 1) included a succession of approximately 30- to 40-min coding sessions with adaptive training levels in effect for each participant. The number of competitive rounds coded per session varied, partly as a function of coding speed and accuracy, but usually included between five to nine rounds. At this stage, criteria for moving up a level were set to 85% accuracy as a running average across 14 consecutive codings, and criteria to move down were set at 65% accuracy. After we deemed the data to be relatively stable across sessions for Participants 1 and 2, we implemented a second experimental condition (Experiment 2). Participant 3 was inadvertently exposed to noncontingent feedback through a programming bug that existed throughout her Experiment 1 conditions, thus causing her to receive what might best be construed as "sham training" during this period. She was moved directly to Experiment 2 conditions when the random feedback functioning was discovered and corrected.

All 3 participants were exposed to altered adaptive control parameters in Experiment 2 in order to assess whether more stringent criteria for moving up (90% accuracy across 20 successive codings) or down (80% accuracy across 20 codings) between adaptive levels might improve accuracy scores and/or reduce variability. After a relatively stable performance level was reached in Experiment 2, the five rounds coded as baseline were coded again (posttraining), and four additional rounds from previously unseen video sources were evaluated as a training generalization condition (generalization). All 3 participants experienced different (nonrandomized) sequences of videos for coding.

Calibrating expected accuracy attainment. Expert systems are only as reliable as the expert model on which they rely for implementation. Variations in alternative videotaping conditions, in the complexity of behavioral phenomena being targeted, and numerous other factors—including the number of categories in a training taxonomy (see Bass, 1987)—can have significant impact on even an expert's consistency in coding the same training events. As such, one may hardly expect trainees to attain accuracy levels consistently higher than an expert's own levels of intraobserver agreement on the same video source materials when the expert is using the same recording procedures as those used by the trainee. In TTC, the expert model is defined by a highly trained individual's codings of all videos, using CC procedures. In the best of circumstances, multiple individuals may collaborate, using constant discussions aimed at establishing a consensus of interpretation and timing to define the expert codings.

In the present case, a single individual expert (J.M.R., who has over a decade of regional and national equine hunter–jumper competition experience) originally coded the entire corpus over a relatively brief span of a few weeks, using CC procedures. Approximately 18 months later, she subsequently coded six training sessions under the same no-feedback (Level 4) BFC training conditions as those experienced by our participants in late sessions. This allowed us to inform our expectations for maximal sustained trainee performances

Table 2
Intraobserver Agreement and Cohen's Kappa Measures for the Expert

Session	Agreement	Kappa
1	.91	.89
2	.91	.88
3	.97	.96
4	.96	.95
5	.95	.94
6	.95	.93
M	.942	.925

Note—Initial codings using continuous procedures are compared with codings made approximately 18 months later using behavioral frequency count procedures. See Figure 6 for plots of round-to-round agreement variability within each of the six sessions reported in this table.

based on same-condition measurements—a procedure we highly recommend for any potential users of TTC.

Results

Data generated from observer codings present numerous problems for defining equivalent value points for comparative purposes. Every round a horse makes in competition includes a variable number of behaviors performed—a phenomenon extant in virtually every naturalistic coding situation involving dynamic behavioral foci, regardless of species, circumstance, or categories. Likewise, the forced error correction procedures of our adaptive tutoring impose a variable number of codings across and within participants even when the same round is coded or a fixed number of behaviors occur. The mean number of total behavioral codings per competitive round used in these experiments was 27.25, based on all 3 participants and all rounds coded. For this experimental series, our most fundamental measure is thus an accuracy measure calculated on a per round basis for each participant. This measure is defined as the number of correct coding inputs divided by the sum of correct codings and all forms of errors made (whether commission or omission) for each participant coding each successive round.

Singular rounds generate relatively few coding entries and, thus, result in a metric that is, in some respects, both very high resolution and very low resolution, as compared with most research using behavioral observation methods. Traditionally, entire coding sessions involving many behaviors are typically used to generate such measures as percentage of agreement or Cohen's kappa. This difference has two implications. First, round-based measures are high resolution in that they are highly sensitive to moment-by-moment fluctuations in coder vigilance and, thus, are likely to reflect much higher within-subjects variability from round to round than is typically reported. The more traditional approach of using much larger coding samples based on entire sessions will smooth such fluctuations dramatically, much as averaging does in any large and variable sample. However, the *mathematical* resolution is actually quite low with very small samples. To take extreme examples to illustrate, a sample of two codings can only vary as 0%, 50%, or 100% accuracy; a sample of three codings varies with intervals of .33, a sample of four varies in intervals of .25, and so forth. Thus, small samples vary in relatively larger intervals and thereby amplify small error differences from round to round.

Second, round-based measurement produces a relatively large number of data points, even when based on a relatively small number of different participants. However, successive within-subjects time series measures involve high degrees of interdependencies, or autocorrelation, thus making standard inferential comparisons based on independent measure assumptions suspect techniques, despite the robust nature of most commonly used tests (Hartmann, 1974; Thoresen & Elashoff, 1974; Toothaker, Banz, Noble, Camp, & Davis, 1983). Although singular *across-session* scores may approximate independent measures by individual, they are likely to include too much of a nonlinear *acquisition* trend in Experiment 1 to allow them to be compared with the assumedly more stable scores in Experiment 2. Thus, in the subsequent section, we will merely illustrate averaged data by condition as a grouped time series. In the section following the group consideration, we will evaluate the individual high-resolution data for each participant considered from the more appropriate and intended design perspective of single-participant time series replications.

Intraobserver reliability for the expert. The evaluation of intraobserver agreements for the expert's BFC-based recoding of a sample of six sessions is summarized in Table 2, and round-by-round variations are illustrated in Figure 6. As with participant codings, each session includes multiple, but varying, numbers of rounds. The columns in Table 2 depict corresponding intraobserver agreement and kappa values for each successive coding session, with the mean and standard deviation across all sessions depicted in the last column. For this expert, there were relatively minor differences between kappa and accuracy

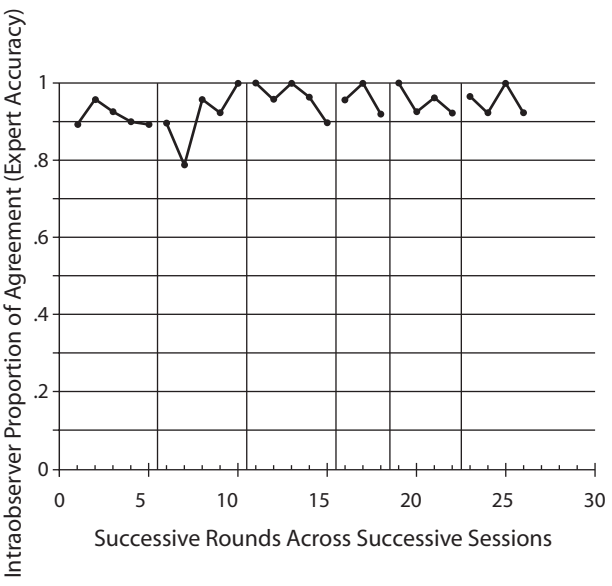

Figure 6. Intraobserver agreement variability in scores across successive rounds of codings within successive sessions using continuous coding procedures versus behavioral frequency count procedures by the expert.

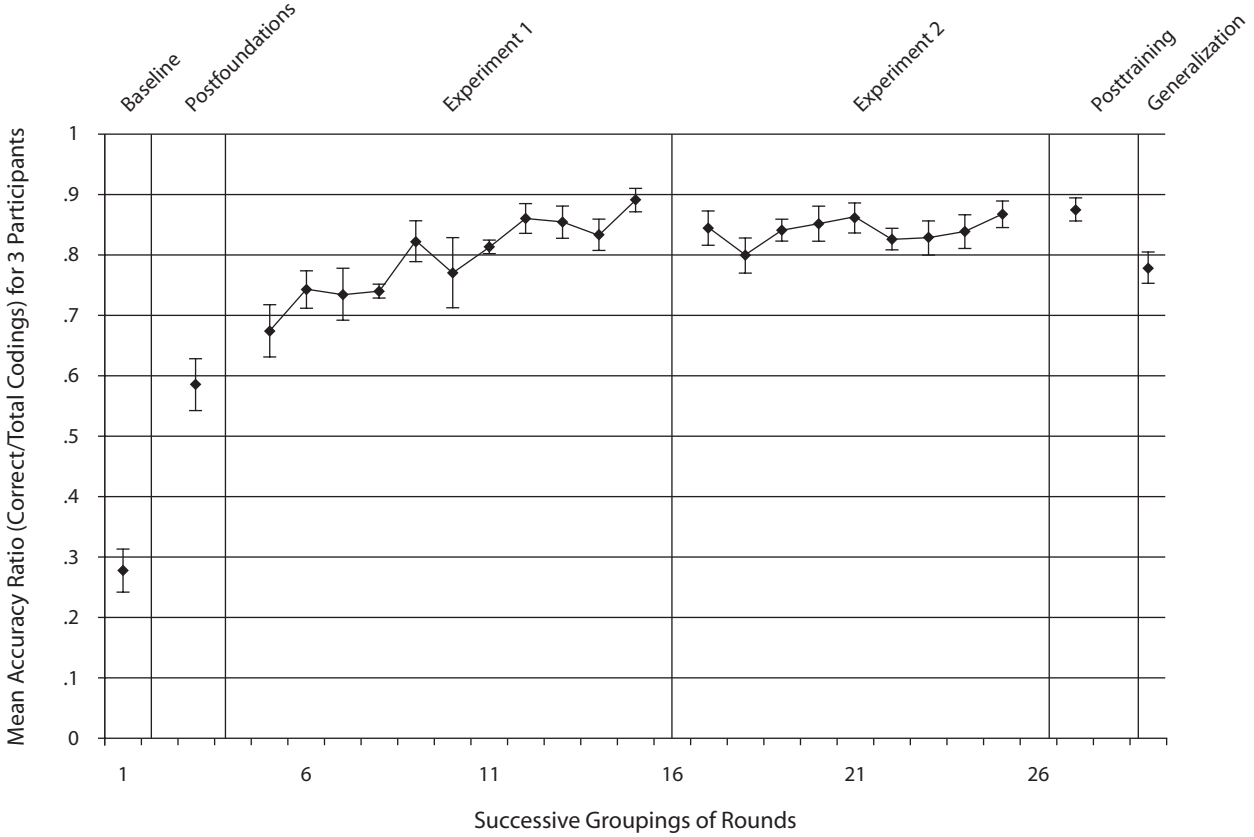

Figure 7. Mean accuracy ratios for all 3 participants combined and calculated across successive five show performance rounds (or fewer, if fewer than five exist) in each experimental condition.

measures, with scores across all sessions ranging from simple percentage-of-accuracy scores of .91–.97 (M = .942) and kappas of .88–.96 (M = .925).

Group perspective. Means and associated standard error scores were calculated to represent experimental phases. Figure 7 illustrates these means as follows. For baseline, all the associated data were combined into one average score. Likewise, one average score represents all the postfoundation rounds. Successions of means for five rounds (or the actual number available, if fewer than five rounds were available for a given participant as training was completed) across all 3 participants represent trends across Experiments 1 and 2. Finally, a single mean represents the follow-up reevaluation of the original baseline rounds, and another represents the generalization condition.

Figure 7 reveals that a collective average of 80% accuracy or better was achieved by the 5th training sample and was sustained in the mid-80% range for all windows between the 7th and the 20th training samples. Standard error ranges reflect that all but one of the averaged scores in each sample include variations that were collectively well within the 80%–90% accuracy range. The final baseline condition was above 85%, but the generalization phase dropped below 80%. These simple interobserver accuracy scores were approximately 8–10 percentage points below the expert's intraobserver accuracy comparison.

Single-participant design perspective. For early system evaluation, our metric based on individual rounds is considered far more useful for informing parametric and/or design adjustments, with the ultimate goal not only of achieving higher individual performance accuracy, but also of minimizing intraparticipant variability. Coding accuracy ratios for each separate round across each condition are thus illustrated in separate graphs for each participant (Figures 8, 9, and 10). To reveal the more stable trends in these data, the figures also include a line representing the mean accuracy across rounds for all rounds within each condition. For the longer experimental conditions, we split each into two halves for early/late training comparisons of trends.

Figure 8 shows that Participant 1 coded with an initial baseline accuracy of 40.5%, using printed reference materials. After initial foundations training, her accuracy increased to 54.8%. Experiment 1 training conditions raised accuracy to 77.5% for the first half of the condition and 86.6% for the second half. A shift to more stringent adaptive parameters resulted in 86.2% and 84.2% accuracy for first and second halves of the conditions, respectively. The posttraining baseline probe had a mean accuracy of 89.8%, which approximated the accuracy scores of our expert in final evaluation conditions. However, the participant dropped to 79.1% in generalization conditions. Also noteworthy was the high variability around these values across all conditions. Experiment 2, for example, included

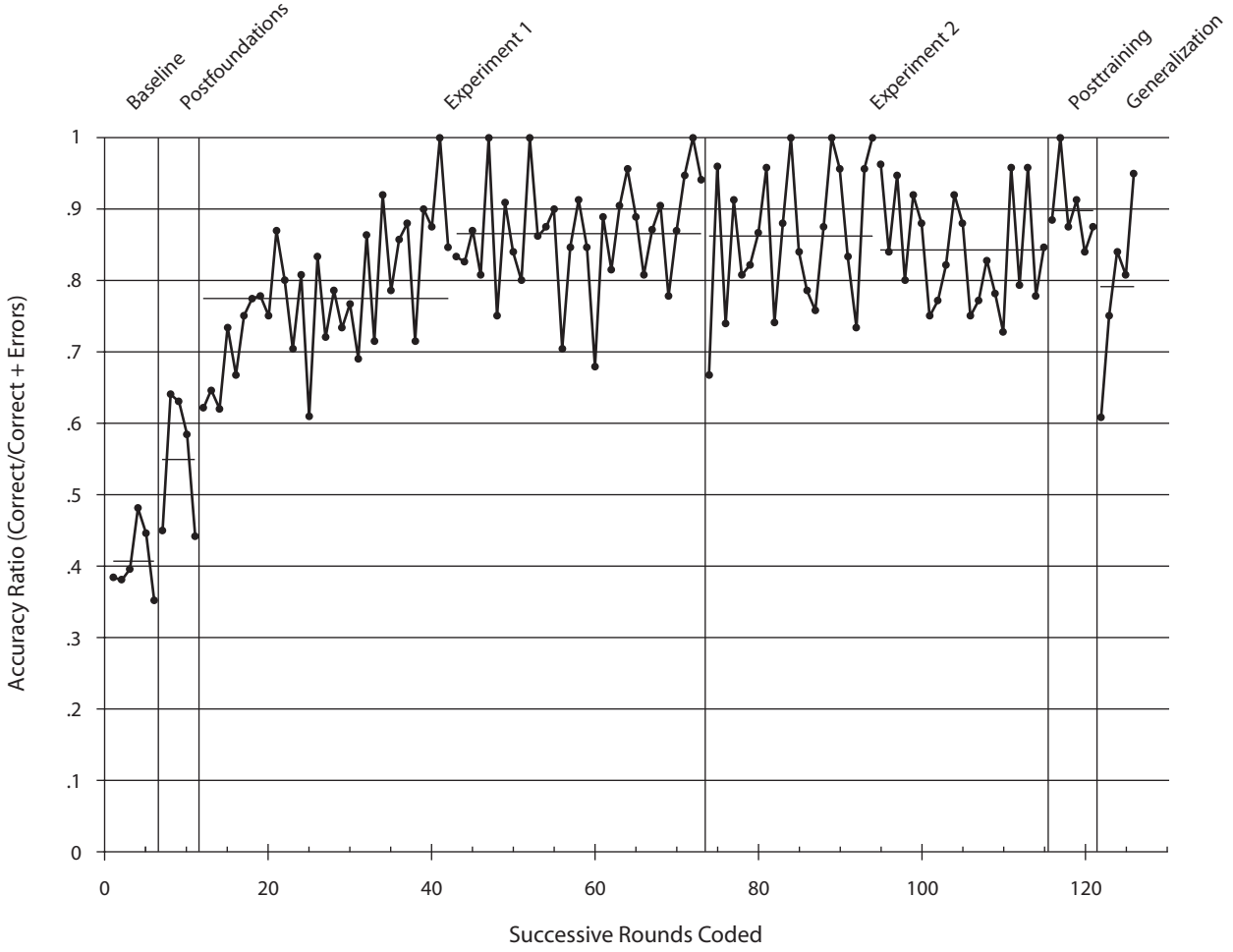

Figure 8. Accuracy ratio for each successive show performance round plotted across all experimental conditions for Participant 1. Means for each experimental condition or split-half conditions are graphed as reference trend lines for individual round plots.

14 rounds coded at or above 90% accuracy, with 3 reaching 100%. On the other hand, 28 rounds were below 90%, with several in the low 70% range. This reflects considerable moment-by-moment variability in coding performance that is hidden by grouped data summaries considered by aggregated rounds.

Figure 9 depicts similar performance levels for Participant 2 across all conditions, with virtually all means at or above 80% accuracy following the beginning of training. Late Experiment 2 and posttraining levels also approached the levels of our expert. Again, the stability of accuracy scores considered round by round was improving by the end of the last half of Experiment 2 but still included substantial variations about the mean.

The noncontingent feedback experienced in Experiment 1 by Participant 3 (see Figure 10) resulted in a mean accuracy below 60%, but accuracy rose to 78.1% and 83.4%, respectively, when the participant was introduced to contingent training and feedback in Experiment 2. This level was essentially maintained during posttraining baseline but dropped to 71.6% in generalization. Again, a prominent feature was the moment-by-moment variability between rounds.

Although none of the participants reached a stabilized level of accuracy at or above 90%, good to excellent *kappa* scores, as defined by Bakeman and Gottman (1997), were achieved by all. Mean kappas calculated across all of Experiment 2 sessions, collapsed for each participant, were .80, .83, and .78, respectively. This suggests that interobserver performance reached a range considered "excellent" according to Bakeman and Gottman's criteria for observers collecting data in a research project. Interobserver agreement (kappa) scores achieved by the participants in Experiment 2 also conform with acceptable research values, using similar numbers of categories reported in peer-reviewed journals such as *Child Development* (e.g., Ross, Ross, Stein, & Trabasso, 2006, reported research kappas of .82 and .85 with 10 categories; Guralnick, Hammond, Connor, & Neville, 2006, reported a kappa of .70 for prestudy reliability, using 10 categories; and Furman & Simon, 2006, obtained kappas of only .72–.78 with 7 categories).

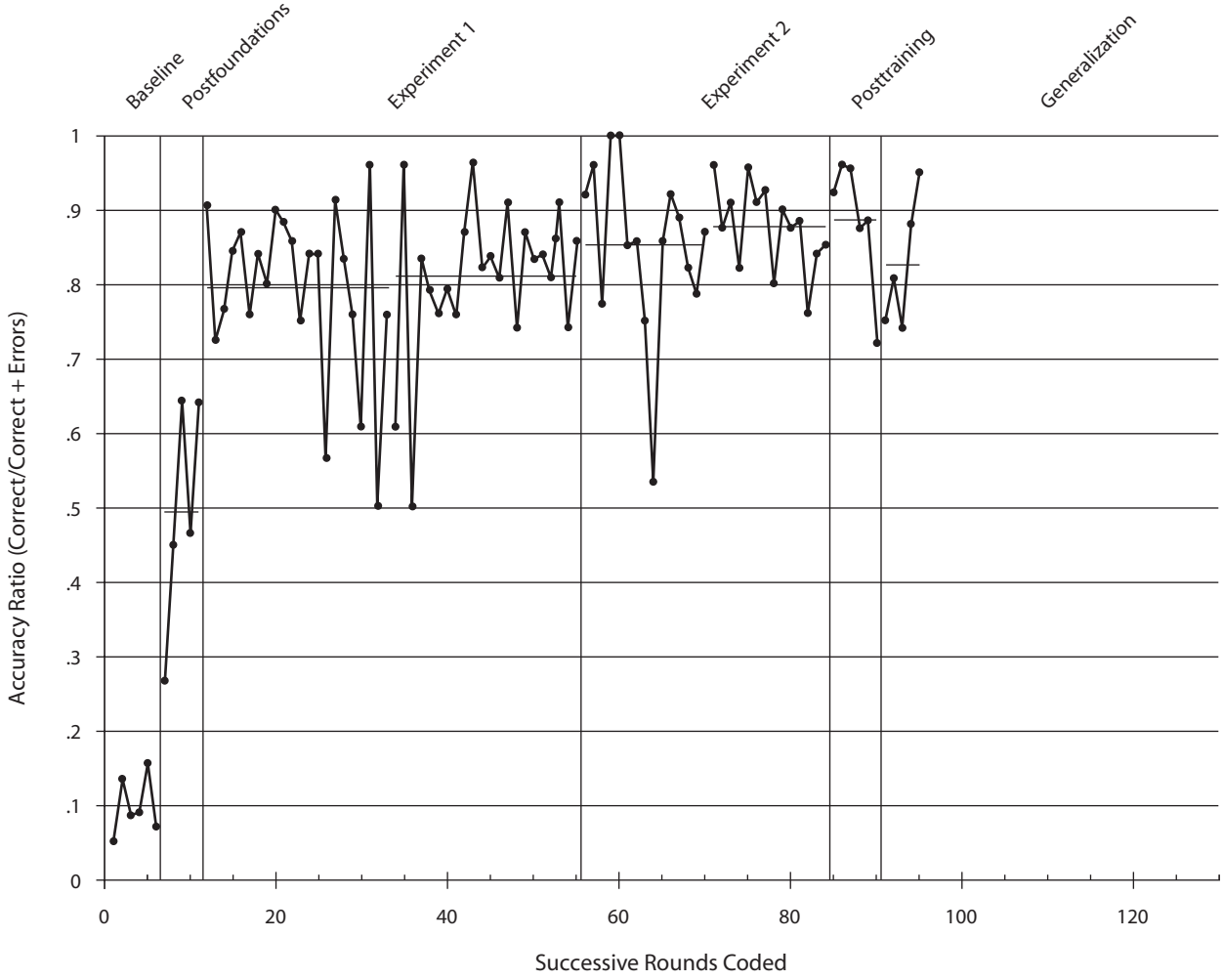

Figure 9. Accuracy ratio for each successive show performance round plotted across all experimental conditions for Participant 2. Means for each experimental condition or split-half conditions are graphed as reference trend lines for individual round plots.

Discussion

In the previous section, we noted a common feature that becomes apparent only with our very high degree of measurement resolution: All 3 participants are quite variable around their respective mean levels. Some of the early variability undoubtedly reflects learning trends, but subsequent variability ranges are quite large as well and warrant further consideration, which we will return to momentarily. It is noteworthy, however, that our expert's stability is substantially higher than our participants', with our expert attaining a mean kappa of .925, whereas the participants attained only .78–.83. Although participant levels appear to reach levels acceptable for our referenced comparative publications, it should be kept in mind that kappa values reflect aggregated variations due to different levels of coding training. Furthermore, coding demands imposed by such factors as types of behavior defining a taxonomy, numbers of categories used, and behavior lengths/durations may impact accuracy.

For TTC formative design purposes, our observed kappas and their associated simple percent-calculated accuracy levels prompted us to explore whether accuracy could be improved via system design changes, since the more stringent parametrics for adaptivity level determination in Experiment 2 had such minimal effect. Because our participants did not stabilize at or above a 90% accuracy level, and because they maintained a substantial variability that occasionally included sizable drops in accuracy ratios, we explored their data for further design-informing feedback.

Fortunately, the TTC's design offers not only in-depth information regarding general accuracy, but also specific information as to the types and frequency of observer errors and the behavioral conditions under which they occur. Table 3 illustrates such a breakdown of types of errors made within two experimental conditions (Experiment 2 and posttraining test). These data reveal that 24.05% (Participant 3, training, Experiment 2) to 55.26% (Participant 2, test) of all errors were *missed behaviors*, whereas 36.07% (Participant 3, test) to 47.33% (Participant 3, training, Experiment 2) of all errors were *mislabeled*. Table 4 illustrates a more detailed breakdown of coding accuracy by behavioral category. This table reveals that

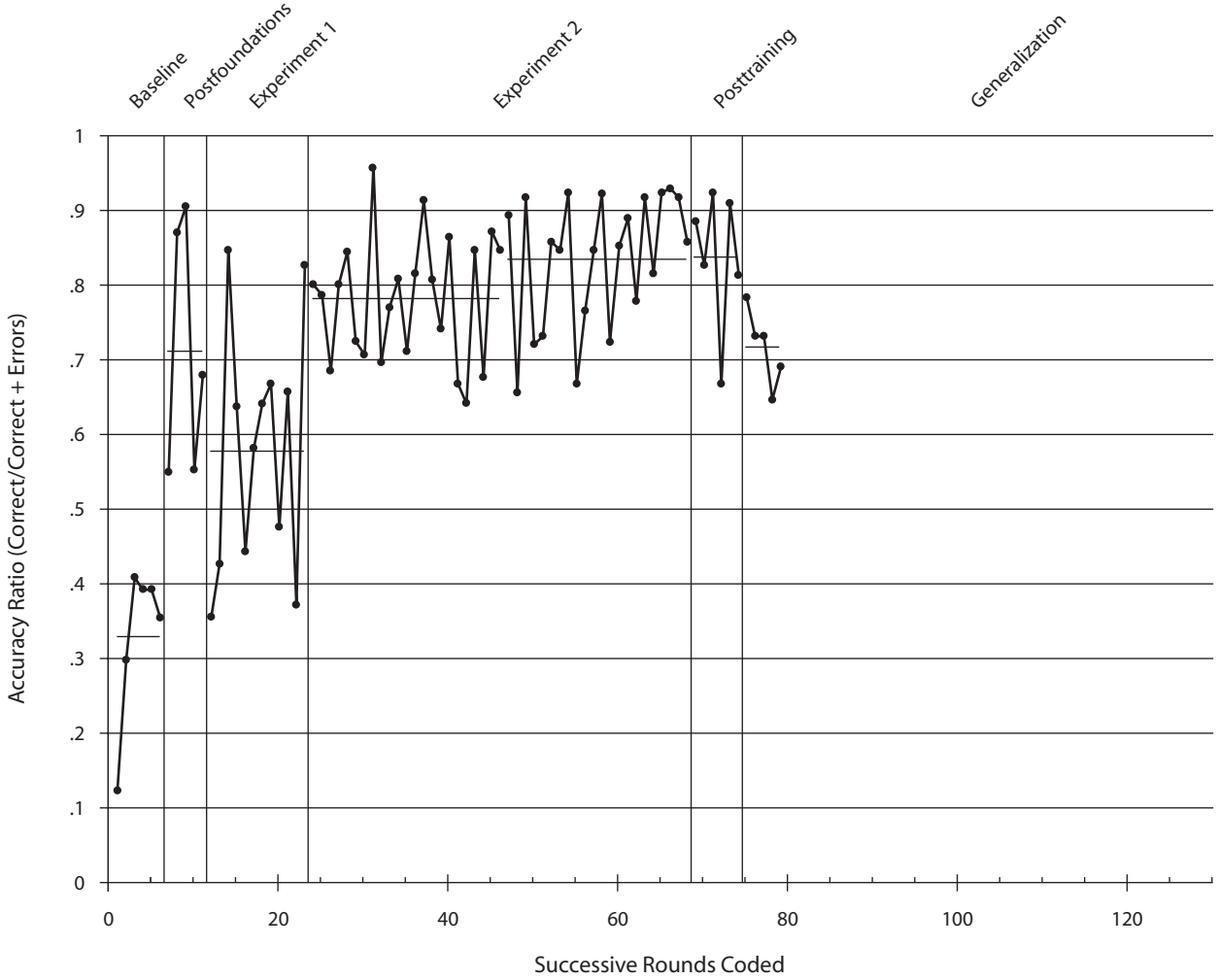

Figure 10. Accuracy ratio for each successive show performance round plotted across all experimental conditions for Participant 3. Means for each experimental condition or split-half conditions are graphed as reference trend lines for individual round plots.

the training system especially failed to teach one difficult-to-discriminate (and often brief-duration) behavior: cross-canter. This particular behavior was coded accurately 40% or less of the time during Experiment 2 and posttraining testing by all 3 participants.

Table 5 illustrates the more common types of errors (not all types) made for each category of behavior. It is clear from this table that the errors made while observing cross-canter were predominantly missed codings. This was especially true of Participants 1 and 2 (from 62.5% to 100% missed, 0% mislabeled), indicating that they simply failed to see the behavior at all, rather than mislabeling it. On the other hand, Participant 3 actually mislabeled this behavior between 33% and 60% of the time, indicating that the behavior was more typically seen (20%–33% missed) but not correctly labeled. Such information is

Table 3
Breakdown of Errors (in Percentages) Across All Behaviors During Training and Final Test Conditions

	Participant 1		Participant 2		Participant 3	
Type of Error	Training Experiment 2	Test	Training Experiment 2	Test	Training Experiment 2	Test
Mislabeled	41.44	38.10	38.18	36.84	47.33	36.07
MultipleC	7.18	2.38	3.63	0.00	16.03	11.48
MultipleW	16.57	4.76	16.36	7.89	12.60	9.84
Missed	34.81	54.76	42.00	55.26	24.05	42.62

Note—MultipleC, multiple coding/correct labeling; MultipleW, multiple coding/wrong labeling.

Table 4
Percentages of Correct Codings by Each Specific Category of Behavior

	Participant 1		Participant 2		Participant 3	
Behavior	Training Experiment 2	Test	Training Experiment 2	Test	Training Experiment 2	Test
JMP	97.07	100.00	97.03	95.35	92.39	78.78
LLC	73.52	76.71	78.26	75.00	69.55	70.00
RLC	75.35	70.15	77.38	81.25	72.29	71.01
TRT	94.59	100.00	97.44	100.00	92.22	94.00
CC	33.33	0.00	12.50	0.00	40.00	29.00
WLK	85.42	66.67	78.38	88.89	79.31	77.78
HLT	0.00	n/a	0.00	n/a	50.00	n/a
CLIP	100.00	100.00	96.55	100.00	88.24	92.00

Note—JMP, jump; LLC, left-lead canter; RLC, right-lead canter; TRT, trot; CC, cross-canter; WLK, walk; HLT, halt.

critical for informing system-wide design elements and quickly resulted in designing changes that focus on error correction and other training features in the higher levels of the training mode.

IMPLEMENTED DESIGN CHANGES

Although these results indicate that our initial system design is relatively effective for training coding accuracy skills, they also suggest the need for minor improvements in the adaptive training features we detailed above. Research-level observer accuracy was achieved quite efficiently, but our failure to require error correction needs to be addressed. As a result of findings from these research probes, we made design changes in most training levels of the student mode. First, we designed all the levels to include feedback for missed behaviors. Second, we designed more stringent error correction requirements for mislabeled and missed behavioral codings. In this revised version, Training Levels 1, 2, and 3 incorporate momentary feedback and replay for incorrectly coded or missed behaviors. Thus, the system now requires that all errors be corrected prior to continuing to the next behavior.

Should a student be confused about a code or begin to guess, which is our implied interpretation after the same behavior has been miscoded three successive times during play/replay, the system shows the correct code along with its definition. It also offers the opportunity to review the

Table 5
Analysis of Mislabeled and Missed Coding Errors
As a Percentage of All Errors Within Each Behavior

	Participant 1		Participant 2		Participant 3	
Behavior	Training Experiment 2	Test	Training Experiment 2	Test	Training Experiment 2	Test
JMP						
Mislabeled	44.44	0.00	43.00	0.00	18.75	18.18
Missed	11.11	0.00	43.00	75.00	6.67	27.27
LLC						
Mislabeled	39.66	47.06	40.00	47.37	47.73	38.10
Missed	34.48	47.06	42.86	52.63	27.27	61.90
RLC						
Mislabeled	49.30	40.00	42.00	41.67	56.19	35.00
Missed	29.58	50.00	32.00	41.67	28.57	35.00
TRT						
Mislabeled	75.00	0.00	0.00	0.00	100.00	0.00
Missed	0.00	0.00	100.00	0.00	0.00	100.00
CC						
Mislabeled	0.00	0.00	0.00	0.00	33.33	60.00
Missed	62.50	100.00	86.00	100.00	33.33	20.00
WLK						
Mislabeled	0.00	0.00	37.50	0.00	41.67	50.00
Missed	57.14	100.00	50.00	100.00	25.00	50.00
HLT						
Mislabeled	100.00	0.00	100.00	0.00	0.00	0.00
Missed	0.00	0.00	0.00	0.00	0.00	0.00
CLIP						
Mislabeled	0.00	0.00	0.00	0.00	16.67	100.00
Missed	0.00	0.00	100.00	0.00	0.00	0.00

Note—JMP, jump; LLC, left-lead canter; RLC, right-lead canter; TRT, trot; CC, cross-canter; WLK, walk; HLT, halt.

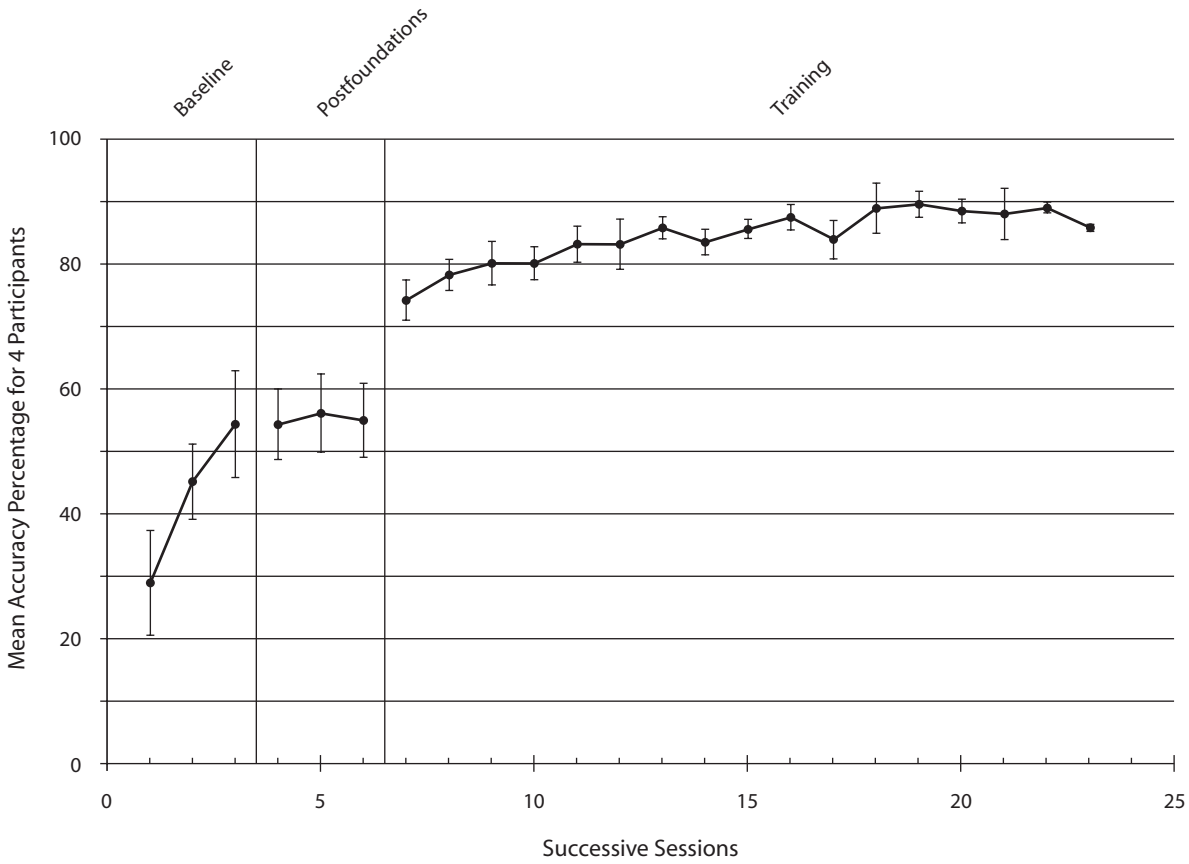

Figure 11. Average scores and their standard errors across successive training sessions for all 4 participants in Experiment 3.

behavior, just as would occur at the foundations levels. This feature also allows the student to replay the behavior for recoding. When a student mislabels or misses a behavior altogether in Training Level 3, the behavior is now replayed with prompting and feedback features identical to those used in Level 2, thus offering the support of a lower level for mislabeled/missed behaviors only. Finally, Training Level 4 now offers a momentary feedback-only (no replay/recode) message for all types of errors, with the type of error being stated in the feedback.

EXPERIMENT 3

Method

Participants. We recruited 4 new female participants between the ages of 18 and 21 years to evaluate the effectiveness of the TTC modifications described above. Each was informed that she was being asked to participate in a longitudinal multisession research design and was expected to return to the laboratory on several different days for participation. The physical conditions described for Experiments 1 and 2 were also used, including a single participant's being seated at a computer training station and being observed by the experimenter from a separate room.

Procedure. All the participants used the same corpus of video source materials coding with our horse gait taxonomy, thus maintaining sports-judging skills as our exemplar application. For this experiment, the original video materials were randomized by individual to avoid any potential order effects and were presented in slightly shorter sessions than in the prior experiments in order to reduce possible vigilance decrement effects. As in previous studies, our BFC coding procedure required the participants to recognize and code each instance of a behavior during its occurrence, including black-screen round separations. Our 4 participants in Experiment 3 received an equal number of training sessions (17), although variations in the number of rounds per session resulted in slight variations in the number of rounds actually coded. This is only slightly less total training than the 20 sessions our previous 3 participants experienced in Experiments 1 and 2 combined.

The procedure replicated that in the first stage of Experiment 1 (baseline) by giving each participant a printed handout with the coding taxonomy and corresponding definitions for each included category. As in Experiment 1, the participants were instructed to study the handout and, subsequently, to use it while coding a series of videos depicting various horses performing a total of six rounds in hunter–jumper competition across three separate coding sessions. In a session following completion of baseline, the participants reviewed all three levels of foundations training. They subsequently coded a randomly sequenced set of videos that included six (for 2 participants) or seven (for 2 other participants) total rounds of competition. These sessions essentially replicated our prior evaluation of the effectiveness of system-based foundations training (postfoundations). A subsequent training condition for Experiment 3 included a succession of 17 coding sessions of three to five competitive show rounds within each session. During all the training sessions, four levels of adaptive training strategies were in effect for each participant, as in Experiments 1 and 2, with the only differences being the TTC design modifications implemented after Experiment 2 as described in our Discussion section above. As in Experiment 2, the criterion for moving up a level was set to 90%

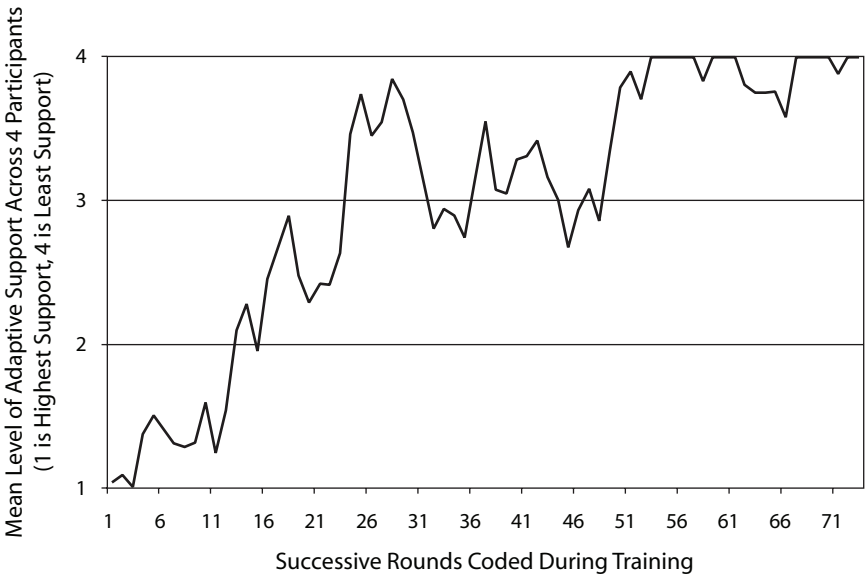

Figure 12. Averaged measures of adaptive level in Train-to-Code by successive rounds coded by the 4 participants in Experiment 3.

accuracy across 20 successive codings, and the criterion to move down was set at 80% accuracy across 20 successive codings.

Results

Grouped data accuracy and stability. Figure 11 depicts grouped scores by successive session for all 4 participants. The baseline phase using only written definitions and code abbreviations shows improvement from 30% up to 55% across three separate sessions, but with substantial variability (standard error) between participants. The experimental phase following exposure to the three foundations levels within TTC show a sustained performance of approximately 55% accuracy, and with only slightly diminished interparticipant variability. Training shows an immediate jump to approximately 75% accuracy during the first session and a gradual increase to levels between 85% and 90% during the last six sessions, with relatively consistent small standard errors for these respective means.

Although we were cognizant of the problems with even simple inferential tests based on assumptions of independent scores, the robustness of a paired-comparisons *t* test led us to explore selected comparisons for statistical information. Because the participants coded a random presentation of session-defining videos twice, once during the first half of training and again during the second half of training, accuracy data for specific *rounds* (vs. sessions) were averaged into first- and second-half scores for each of the 4 participants. An increase in accuracy from the first half of training (81.4% mean accuracy across all rounds and all participants) to the second half of training (87.5%) was evaluated using SPSS to conduct a *t* test for paired data. Statistically reliable differences were obtained [$t(3) = 6.709$, $p < .01$], indicating that the group significantly improved accuracy performance from the first half to the second half of the experiment, as one would expect of training interventions.

Design modifications preceding Experiment 3 focused on efforts to increase accuracy scores while also decreasing variability in those scores. To explore the effectiveness of these design modifications, the single mean scores and single standard deviations for accuracy scores representing the last 30 rounds of coding for each of the 3 participants in our earlier Experiment 2 were compared with the same calculations for the 4 participants in Experiment 3. Although the means increased from 84.7 in Experiment 2 to 87.7 in Experiment 3, a *t* test based on individual mean accuracy scores showed that differences between independent groups were not significantly different [$t(5) = 1.215$, n.s.]. To evaluate the relative stability of the last 30 accuracy scores for each participant in Experiment 2 versus Experiment 3, a separate analysis treated the standard deviations for the means above for each participant as the raw score. Even though group sizes for this comparison were again only 3 and 4 participants, respectively, a *t* test for differences between the standard deviation scores resulted in $t(5) = 4.183$, $p < .01$, thus indicating that variability in scores was significantly lowered for the participants in Experiment 3.

Training aspects. One noteworthy aspect of the data in both Experiment 1 and Experiment 3 is the relatively small degree of required training at the outset. Whereas participants typically attained approximately 55%–60% accuracy simply from watching examples and reading definitions during foundations training, their very first sessions of real-time training reflected jumps to 65%–75% accuracy within the first session. Given that they attained maximum levels of accuracy in the 80%–90% range, this reveals what might appear as relatively small acquisition/improvement effects even over extended training sessions.

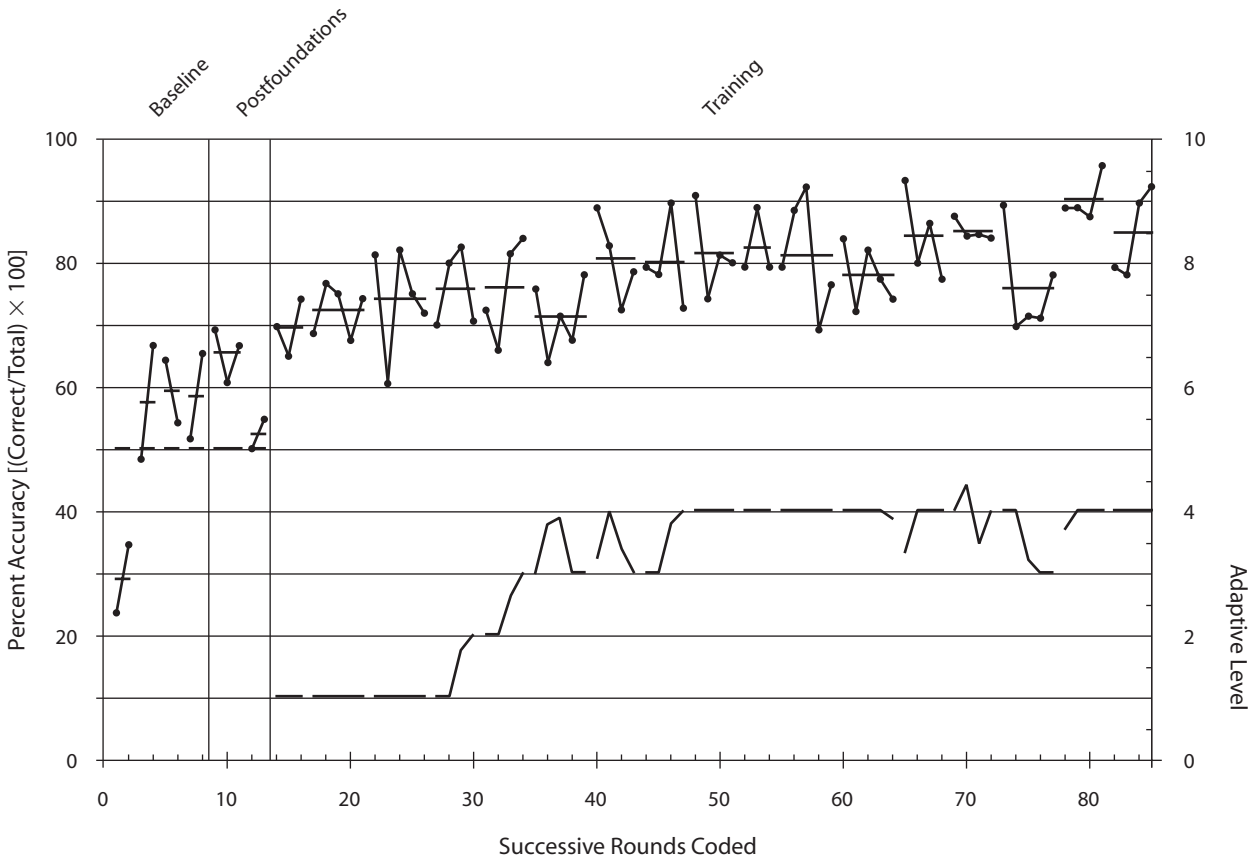

Figure 13. Single-participant accuracy and training service levels by sessions and by rounds within sessions for Participant 05.

It is critical to consider whether accuracy levels include the support services or not. Only nonprompted coding would ever be considered as research-level skills. To assess whether our adaptive training strategy would account for this sustained high level of performance from the outset, we investigated the average *support* level in effect across the various sessions reflected in the training accuracy graph (i.e., Figure 11). Each adaptive training level was given a numeric value of 1, 2, 3, or 4, respectively, with Level 1 incorporating the *highest* degree of support and Level 4 the *lowest* degree of support. These values were then averaged across our 4 participants in Experiment 3 within each competitive round coded. The results of this analysis are depicted in Figure 12, which reveals where the most substantial *learning curve* effect occurred during this training experiment. That is, TTC is designed to be highly supportive with prompts and feedback in early stages of training and is sufficiently accomplished in this goal as to ensure initial training levels of 65%–75% accuracy, but with mostly Level 1 (highest degree) support. There is minor improvement in the accuracy across sessions, but a very clear and systematic withdrawal of prompting and feedback support for that sustained performance. By the last quarter of training, there is only sporadic support required, and that being of a minimalist nature.

Single-participant perspective. As with our prior experiments, our present experiment may also be viewed as multiple within-subjects measures across multiple (four) replications of a single-participant research design. Thus, Figures 13–16 depict single-participant accuracy levels by sessions and by rounds within sessions for each participant in Experiment 3. These reflect individual attainments in both accuracy (session levels, graphed as lines) and stability (rounds within sessions, graphed by dots). Also graphed against the right-hand ordinate is the individualized data on adaptive training support levels that correspond to respective rounds being coded. These adaptive support levels remain inversely associated with their numbers, with minimal prompting and feedback being given in Level 4 and maximal support being given in Level 1.

Detailed inspection of these four figures reveals the true dynamics of the training system and the individualization of its services in almost instantaneously developing high accuracy scores and differentially responding to momentary drops in individual accuracy. Participants 05 and 08 learned to code with virtually total (Participant 08) or nearly total (05) sustained fading of all supportive prompting and feedback. That is, both retained running averages above the 80% trigger level that recommends a drop in training level to reinstate higher degrees of prompting and training support. Participant 08 stayed at Level 4's highly faded services consistently throughout Sessions 9–17. Participant 05 attained the minimal prompting and feedback of Level 4 by the 9th session.

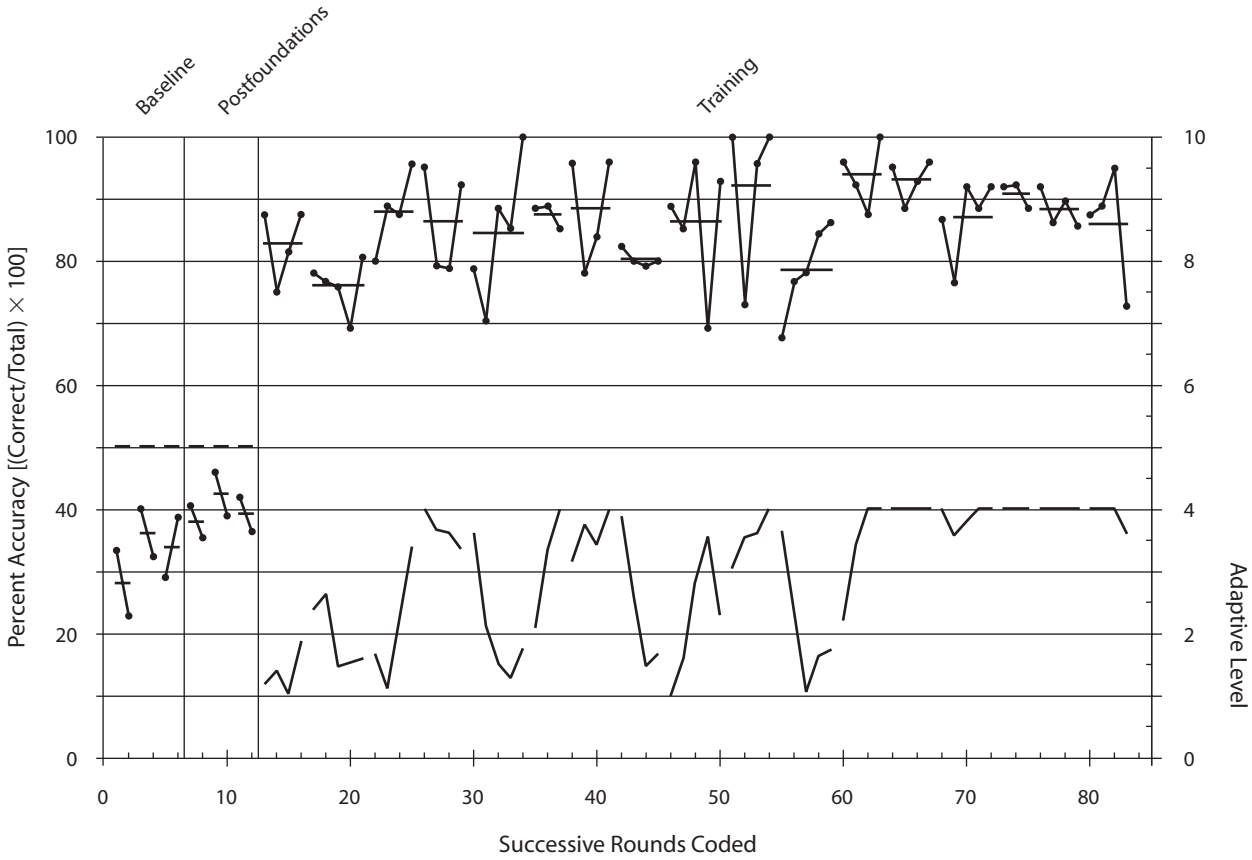

Figure 14. Single-participant accuracy and training service levels by sessions and by rounds within sessions for Participant 06.

She later experienced three brief returns of higher level support services, however, with the most support being returned in the latter portions of Session 15. Participants 06 and 07, on the other hand, were much more variably reliant upon training interventions and did not reach relatively sustained Level 4 services until their last 5 or 6 sessions. In fact, Participant 07 dropped all the way to Level 2 services, which are highly prompted and supportive, in the final two rounds of her 17th session. Nevertheless, session averages for accuracy stayed consistently at acceptable research standards throughout the last 5 sessions of training for both Participants 06 and 07.

GENERAL DISCUSSION AND CONCLUSIONS

The results from the preliminary formative experiments that we present indicate that TTC and, especially, the adaptive operant training model it incorporates are effective for training novice observers to accurately recognize and label almost all behaviors, using the taxonomy and video content employed. Although the first two experiments involved some limitations, our third experiment addressed several critical issues raised. For example, the statistically reliable decrease in intraindividual variability achieved in Experiment 3 is encouraging, in that it provides support for the design modifications made following the first series of experiments. It demonstrates the valuable role of continuing formative research.

How this increased stability was achieved is especially noteworthy. All too often, researchers report only accuracy or interobserver agreements (whether measured by simple percentage-of-agreement or kappa scores) for entire experiments or perhaps, at best, across specific samples of observer evaluation sessions. As we suggested in the opening paragraph to this article, it is extremely rare to find reports of intraobserver accuracy stability in observational research. Our statistical evaluation for the 4 participants in Experiment 3, when considered as a group, demonstrates reliable differences in standard deviations between those 4 participants and the 3 participants in Experiments 1 and 2 across their final 30 rounds of coding. But there is little suggestion of within-session decreases of variability throughout training for the individual participants in Experiment 3.

Although there is a slight, but statistically reliable, increase in accuracy from the first half of training to the second half of training in Experiment 3, these data alone reflect little apparent learning curve on a scale most learning researchers are accustomed to seeing. That is, all 4 participants began at very high levels of accuracy (70%–83%) in their first training session and only very gradually achieved more sustained levels of accuracy any higher than this. So where is the learning curve for these

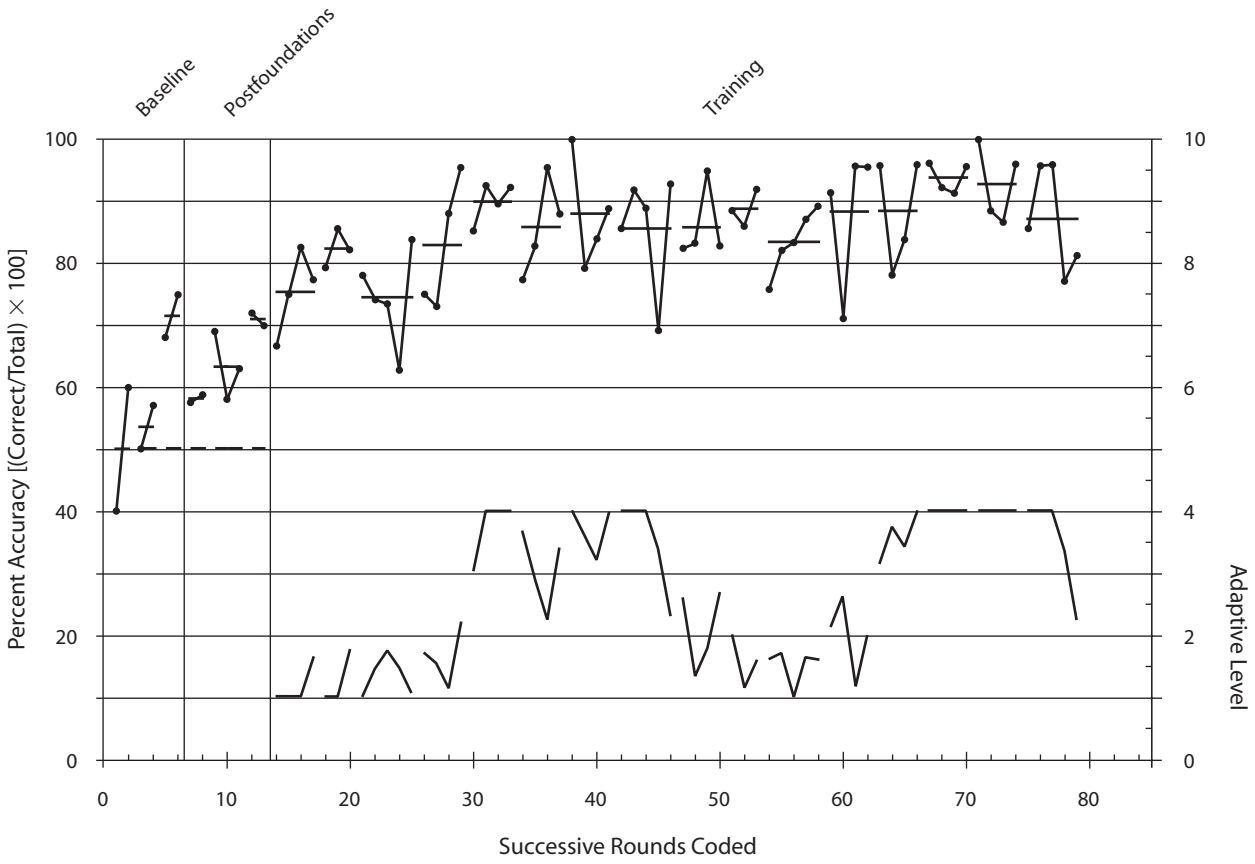

Figure 15. Single-participant accuracy and training service levels by sessions and by rounds within sessions for Participant 07.

participants? As we have clearly illustrated in Figure 11, it is in the fading of supportive prompting that is being used to shape discriminative coding behaviors. That is, the shift from Level 1 through Levels 2, 3, and finally 4 appears much more like a learning curve than do the very gradual changes in accuracy. This is a phenomenon commonly recognized by operant researchers as *errorless discrimination learning* (see Sidman & Stoddard, 1967; Terrace, 1963a, 1963b). And it has suggested to us yet another design modification to enhance this phenomenon and to build upon the effects of foundations training.

The substantial gain in accuracy following mere exposure to our foundations levels has suggested that we might well have the opportunity to develop an even more efficient analogue to operant errorless discrimination procedures. In such procedures, participants are sufficiently prompted as to virtually guarantee successful discrimination from the outset of training. Subsequently, these obvious prompts are faded gradually in successive approximations to the eventual generalization goal, thus allowing new discriminations to develop while the participant makes very few, if any, errors in responding to the emergent discriminative nature of relevant stimuli. Drawing on this added operant instructional design dimension, we are currently modifying TTC to meld our existing two "advanced," but passive, levels of foundations services into a new lower level, or more supportive, dual-service training level.

Existing foundations services allow participants to play a video while seeing the label for the behavioral events being illustrated. We are incorporating this *labeled play* of the video, in a slightly modified form, within our training services. We are going to require a trainee to engage in all of the coding input and system control (play, pause, replay, etc.) behaviors that advanced training levels currently rely on. However, we also provide sufficient prompts (including a clear "labeling" of the behavior as to how it should be coded, as well as which keypad equivalent to use for its coding) to start participants with as few errors as possible. From this *obvious labeling and instructions* stage, we can then fade (see Sidman & Stoddard, 1967) prompts to service levels that systematically provide presentations of less obvious stimulus conditions. The goal of such a change is to bridge the gaps between the 50% performance observed in postfoundations versus the 75% performance observed during initial training.

Other planned features do not really require research so much as timely implementation. For example, in its current development state, TTC saves all data locally as files. Once we are satisfied with the efficacy of the system and have solidified decisions on all data structures, we plan to implement TTC as a server-driven database system. In this configuration, the application and video will exist on the client side, conversing via the Internet with server-stored expert and coding data. This model mirrors

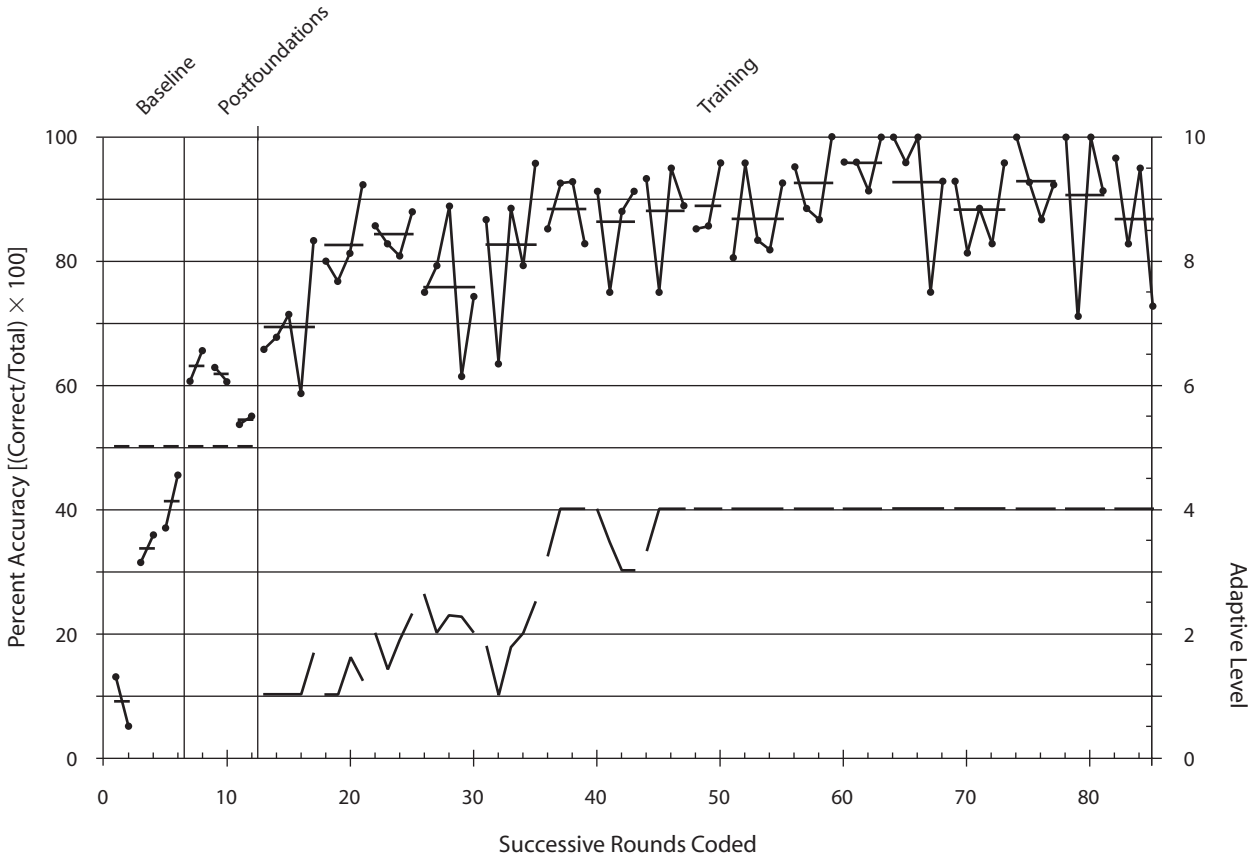

Figure 16. Single-participant accuracy and training service levels by sessions and by rounds within sessions for Participant 08.

our other applications, such as CyberRat (R. D. Ray, 2003) and MediaMatrix (R. D. Ray, 2004).

Future studies are needed to investigate effective uses for the code viewer feature, as well as a newly incorporated notes system that allows an instructor/expert to highlight coding decision criteria for specific problematic behavioral instances. We anticipate that this notes feature will especially help in cases in which coding criteria are subtle or potentially confusing, as they frequently were with cross-cantering in the study above. Notes should also be a significant aid where video events obscure various, and often critical, features of behaviors, thus requiring inferences to be applied by the observer. Under these conditions, as an expert codes any given reference file, notations can be made as to why a specific coding decision was made. This notation will then be accessible in the code viewer to help the student more thoroughly review and understand errors and applicable criteria prior to further training.

There are many issues relevant for future deliberation that are apparent to us as we continue TTC development. For example, in our experiments, a limited corpus of video illustrations required randomized reuse of prior videos during the second half of each experiment. We are fairly confident that familiarization with a video was not a significant factor, given the sheer volume of rounds shown, as well as the substantial number of days extant between early sessions and subsequent sessions in our within-subjects longitudinal experimental designs. Nevertheless, when one attempts to train new participants, the question needs to be considered as to whether reuse of any video source materials is desirable or not. We think a case can be made in favor of overtraining on any given video materials, to perhaps allow one to become more attentive to unique and occasionally inconspicuous discriminative features of specific behavioral events in hopes of gaining generalization to new video source materials. Such issues suggest future research opportunities to evaluate whether, in applied settings especially, multiple video reuse is desirable or not desirable.

Although we believe TTC has utility for training observational coding skills in the classroom and laboratory, this was not the only intent in its design. The program was primarily developed as a training tool for workers in applied settings. TTC focuses on (1) teaching the discriminative recognition of behavior, (2) training arbitrary verbal labels for behavior (what operant researchers refer to as *tacting*; see Skinner, 1957), and (3) given appropriately defined response-focused taxonomies, even the appropriate timing and identification of a desired interventional or social response to specific classes of behavior. The tool was not designed exclusively for the development of coding schemes, nor for the continuous coding of behaviors of focal interest, although these stages are required for expert setups.

What TTC *was* designed for is highly efficient and effective training of users with respect to their ability to code any desired corpus of behavioral examples at a maximal standard of accuracy and stability. As such, TTC is a flexible tool specifically designed for use in the applied setting, as well as in the classroom and laboratory—unlike other currently available systems, such as COR (Blasko et al., 1998) and OBSERVE (Dickins et al., 2000), which tend to target academic settings in their pedagogy (but certainly not in their data collection capabilities).

Thus, TTC should easily find use in a variety of applied settings, from residential behavioral treatment facilities, to zoos, childcare, and judging sports or artistic performances. All that seems required is the development of appropriate stimulus materials (video and taxonomies). We would especially stress that, where appropriate, functionally oriented taxonomies (as opposed to the structurally oriented taxonomy we have illustrated) that describe *desirable interventions*, rather than an observed subject's movements or form, may be highly desirable alternatives as a first-stage effort to train actual intervention decisions.

Our own initial venture into the use of functional categories of behavior is in progress and has already led us to recognize a major procedural difference that has now been incorporated into the current version of TTC. Structural behaviors, such as sit or stand, are quite easy to detect almost immediately upon their initiation. We can easily see and record that an observed subject is standing, sitting, or using required materials for a task. We can discriminate and categorize any other *structurally defined* event well before it ends, thus leading to the procedure (and prompt designs) incorporated into the experiments herein reported. But functional behaviors, including most verbal utterances, typically cannot be classified until after they have been completed—or until they have accomplished the functional goal that defines their purpose. Thus, a behavior such as *turning out the light* or *asking a clarification question* must be coded after the behavior is complete—and typically, while some subsequent behavior is ongoing. This implies that our present use of the "early/middle/late" framing prompts, as well as the "missed behavior" error, requires a totally different strategy and service implementation for functional behavior coding. As such, TTC now has a set of mechanics for training observers to code structural behavior frequency counts (SBFCs) and quite a different set for coding functional behavior frequency counts (FBFCs). We are presently pursuing evaluative probes into the use of such alternative taxonomies, alternative focal observation subjects, and alternative TTC procedural dynamics, including those suited to social, linguistic, and functional taxonomies applicable in both research and applied settings.

Of special interest to us is the degree to which training to code and, thus, to accurately discriminate various behaviors also will generalize so as to enable trainees to actively engage in the discriminated behaviors themselves. Certainly, the vast literature on vicarious/observational learning (see Bandura, 1986; Catania, 1998) leads one to expect such a generalization. One example we are currently pursuing is the investigation of whether we can teach undergraduate college students how to scaffold prereading book-and-story interaction behaviors in preschool children in a developmental laboratory school. We are filming expert scaffolding examples and have created a taxonomy for coding various alternative strategic types of verbal scaffolding. We are now exploring whether training students to code such events will also teach those same students more quickly and with less instructor interventions to use such scaffolding techniques in their own lab-school interactions with the preschool children. Current pilot data are highly encouraging and suggest that our instructional design strategy and TTC modifications for adding FBFC procedures are not only valid, but also sufficiently effective in application to make the TTC system well worth the time and effort being invested in its development.

AUTHOR NOTE

Train-to-Code is a software system developed under the auspices and support of $(AI)^2$, Inc., a software research, development, and publishing company in which both authors are principals. Experiments 1 and 2 were conducted as an Honors in the Psychology Major Senior Thesis by J.M.R. while a senior undergraduate at Rollins College. She thanks her thesis director, John M. Houston, and her committee members, Maria R. Ruiz and Paul B. Harris. A paper based on that thesis was presented at the 2006 Meeting of the Society for Computers in Psychology and received the Castellan Student Paper Award. Experiment 3 was conducted by both of the present authors, and we acknowledge critical data collection contributions by Jessica Crown. Experiment 3 was the basis of a paper presented at the 2007 Meeting of the Society for Computers in Psychology. Correspondence concerning this article and requests for reprints should be sent to R. D. Ray, Department of Psychology, Rollins College, Winter Park, FL 32789 (e-mail: rdray@rollins.edu).

REFERENCES

Andronis, P. T., Twyman, J. S., & Stikeleather, G. (2004, August). *Headsprout early reading: The proper use of formative and summative evaluation for determining the success of behavioral programs*. Panel discussion presented at the meeting of the Association for Behavior Analysis International, Campinas, Brazil.

Bakeman, R., & Gottman, J. M. (1997). *Observing interaction: An introduction to sequential analysis* (2nd ed.). Cambridge: Cambridge University Press.

Bandura, A. (1986). *Social foundations of thought and action: A social cognitive theory*. Englewood Cliffs, NJ: Prentice Hall.

Bass, R. F. (1987). Computer-assisted observer training. *Journal of Applied Behavior Analysis*, **20**, 83-88.

Blasko, D. G., Kazmerski, V. A., Corty, E. W., & Kallgren, K. A. (1998). Courseware for observational research (COR): A new approach to teaching naturalistic observation. *Behavior Research Methods, Instruments, & Computers*, **30**, 217-222.

Blasko, D. G., Kazmerski, V. A., & Torgerson, C. N. (2004). COR V2: Teaching observational research with multimedia courseware. *Behavior Research Methods, Instruments, & Computers*, **36**, 250-255.

Catania, A. C. (1998). *Learning* (4th ed.). Upper Saddle River, NJ: Prentice Hall.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational & Psychological Measurement*, **20**, 37-46.

Dickins, D. W., Kwint, M. A. C. G., Magnusson, M. S., Neads, C. M., Noldus, L. P. J. J., & Quera, V. (2000). OBSERVE: A multimedia course on the observational analysis of behavior. *Behavior Research Methods, Instruments, & Computers*, **32**, 263-268.

Furman, W., & Simon, V. A. (2006). Actor and partner effects on adolescents' romantic working models and styles on interactions with romantic partners. *Child Development*, **77**, 588-604.

Graesser, A. C., Jackson, G. T., & McDaniel, B. (2007). AutoTutor holds conversations with learners that are responsive to their cognitive and emotional states. *Educational Technology*, **47**, 19-22.

Graesser, A. C., Lu, S., Jackson, G. T., Mitchell, H. H., Ven-

TURA, M., OLNEY, A., & LOUWERSE, M. M. (2004). AutoTutor: A tutor with dialogue in natural language. *Behavior Research Methods, Instruments, & Computers*, **36**, 180-192.

GURALNICK, M. J., HAMMOND, M. A., CONNOR, R. T., & NEVILLE, B. (2006). Stability, change, and correlates of the peer relationships of young children with mild development delays. *Child Development*, **77**, 312-324.

HARTMANN, D. P. (1974). Forcing square pegs into round holes: Some comments on "An analysis-of-variance model for the intrasubject replication design." *Journal of Applied Behavior Analysis*, **7**, 635-638.

JOHNSTON, J. M., & PENNYPACKER, H. S. (1993). *Strategies and tactics of behavioral research* (2nd ed.). Hillsdale, NJ: Erlbaum.

LAYNG, T. V. J., STIKELEATHER, G., & TWYMAN, J. S. (2006). Scientific formative evaluation: The role of individual learners in generating and predicting successful educational outcomes. In R. F. Subotnik & H. J. Walberg (Eds.), *The scientific basis of educational productivity* (pp. 29-44). Greenwich, CT: IAP.

LEHNER, P. N. (1998). *Handbook of ethological methods* (2nd ed.). Cambridge: Cambridge University Press.

MARTIN, P., & BATESON, P. (1993). *Measuring behaviour: An introductory guide* (2nd ed.). Cambridge: Cambridge University Press.

PATTERSON, G. R., & MOORE, D. (1979). Interactive patterns as units of behavior. In M. E. Lamb, S. J. Suomi, & G. R. Stephenson (Eds.), *Social interaction analysis: Methodological issues* (pp. 77-96). Madison: University of Wisconsin Press.

PETERSON, G. B. (2004). A day of great illumination: B. F. Skinner's discovery of shaping. *Journal of the Experimental Analysis of Behavior*, **82**, 317-328.

RAY, J. M., GILLINS, L. J., & RAY, R. D. (2006, March). *An operations analysis framework for classifying alternative observational sampling and recording procedures*. Paper presented at the Meetings of the Southeastern Psychological Association, Atlanta.

RAY, R. D. (1995). A behavioral systems approach to adaptive computerized instructional design. *Behavior Research Methods, Instruments, & Computers*, **27**, 293-296.

RAY, R. D. (2003). *CyberRat* (Version 2.0). Winter Park, FL: $(AI)^2$, Inc.

RAY, R. D. (2004). Adaptive computerized educational systems: A case study. In D. Moran & R. Mallott (Eds.), *Evidence-based educational methods* (pp. 143-170). San Diego: Academic Press.

RAY, R. D., & BELDEN, N. (2007). Teaching college level content and reading comprehension skills simultaneously via an artificially intelligent adaptive computerized instructional system. *Psychological Record*, **57**, 201-218.

RAY, R. D., GOGOBERIDZE, T., & BEGIASHVILI, V. (1995). Adaptive computerized instruction: Learning what students learn while learning, teaches computers to improve teaching while teaching. In *Proceedings of the Orlando Multimedia '95 Conference.* Warrenton, VA: Society for Applied Learning Technology.

ROSS, H., ROSS, M., STEIN, N., & TRABASSO, T. (2006). How siblings resolve their conflicts: The importance of first offers, planning, and limited opposition. *Child Development*, **77**, 1730-1745.

SACKETT, G. P. (ED.) (1978). *Observing behavior* (Vol. 2). Baltimore: University Park Press.

SHARPE, T., & KOPERWAS, J. (2003). *Behavior and sequential analyses: Principles and practice*. Thousand Oaks, CA: Sage.

SIDMAN, M., & STODDARD, L. T. (1967). The effectiveness of fading in programming a simultaneous form discrimination for retarded children. *Journal of the Experimental Analysis of Behavior*, **10**, 3-15.

SKINNER, B. F. (1957). *Verbal behavior*. New York: Appleton-Century-Crofts.

TAPLIN, P. S., & REID, J. B. (1973). Effects of instructional set and experimenter influence on observer reliability. *Child Development*, **44**, 547-554.

TERRACE, H. S. (1963a). Discrimination learning with and without "errors." *Journal of the Experimental Analysis of Behavior*, **6**, 1-27.

TERRACE, H. S. (1963b). Errorless transfer of a discrimination across two continua. *Journal of the Experimental Analysis of Behavior*, **6**, 223-232.

THORESEN, C. E., & ELASHOFF, J. D. (1974). "An analysis-of-variance model for intrasubject replication design": Some additional comments. *Journal of Applied Behavior Analysis*, **7**, 639-641.

TOOTHAKER, L. E., BANZ, M., NOBLE, C., CAMP, J., & DAVIS, D. (1983). $N = 1$ designs: The failure of ANOVA-based tests. *Journal of Educational Statistics*, **8**, 289-309.

VAN DER MOLEN, H. H., KERKHOF, J. H., & JONG, S. A. (1983). Training observers to follow children and score their road-crossing behaviour. *Ergonomics*, **26**, 535-553.

(Manuscript received November 19, 2007;
revision accepted for publication March 12, 2008.)