Educational Measurement
ISSUES AND PRACTICE

NCME national council on measurement in education

Educational Measurement: Issues and Practice
Fall 2014, Vol. 33, No. 3, pp. 36–46

Managing What We Can Measure: Quantifying the Susceptibility of Automated Scoring Systems to Gaming Behavior

Derrick Higgins, *Civis Analytics,* and Michael Heilman, *Educational Testing Service*

As methods for automated scoring of constructed-response items become more widely adopted in state assessments, and are used in more consequential operational configurations, it is critical that their susceptibility to gaming behavior be investigated and managed. This article provides a review of research relevant to how construct-irrelevant response behavior may affect automated constructed-response scoring, and aims to address a gap in that literature: the need to assess the degree of risk before operational launch. A general framework is proposed for evaluating susceptibility to gaming, and an initial empirical demonstration is presented using the open-source short-answer scoring engines from the Automated Student Assessment Prize (ASAP) Challenge.

Keywords: artificial intelligence, automated scoring, constructed response, machine learning, simulation

Introduction

Since the very earliest work on automated scoring of constructed-response (CR) items, evaluations of efficacy have focused on how well the scores assigned by the system agree with scores assigned independently by human judges. This was the evaluation criterion pursued in Ellis Page's initial investigations into methods for automated scoring of essays (Page, 1968), and some form of agreement measure (whether Pearson correlation, a form of Cohen's kappa, or a simple percentage agreement) has been used in almost every research paper published on automated scoring methodology (e.g., Attali, Bridgeman, & Trapani, 2010; Bernstein, Van Moere, & Cheng, 2010; Landauer, Laham, & Foltz, 2003; Page, 1994; Zechner, Higgins, Xi, & Williamson, 2009).

Of course, agreement with human scores is one component in a validity argument supporting its use in an assessment, but it is not the only measure relevant to determine the proper role of the technology. One other important criterion for automated CR scoring systems is that they do not introduce unexpected and pernicious differential effects across subgroups of interest (cf. Bridgeman, Trapani, & Attali, 2009). Another criterion is that their scoring be demonstrably grounded in factors related to the construct targeted by the item (cf. Quinlan, Higgins, & Wolff, 2009). One reason why such criteria may have been neglected in previous work is that they are more difficult to quantify. Whereas agreement with human raters can be simply calculated using existing score data, other criteria can be difficult to operationalize or may require metadata (such as test taker demographics) that are not universally available. A related reason may be that because of the difficulty of measuring performance on criteria other than agreement, assessment programs may see these other criteria as logically secondary. (That is, until the system's agreement with human raters has been established, there is no point in investing resources to conduct detailed analyses of fairness or construct alignment.)

Another important criterion, acknowledged obliquely in many studies but never explicitly quantified (cf. Williamson, Xi, & Breyer, 2012), is that automated CR scoring systems should not be susceptible to construct-irrelevant "gaming" strategies, by which test takers may seek to inflate their scores. This paper aims to remedy the lack of attention that this particular criterion for use of automated scoring systems has received, by providing a measure (or at least a framework for such measures) that can be used to benchmark and improve systems' susceptibility to gaming strategies.

Automated Constructed-Response Scoring and Validity Research

The threat that application of construct-irrelevant response strategies to automatically scored CR items poses to test validity has long been recognized. However, previous research has not attempted to develop methods of quantifying and comparing the susceptibility of different scoring methods to such strategies. There is nonetheless substantial research in a number of areas that informs the approach taken here.

A few studies have investigated the strategies that are popularly believed to be effective in influencing the score assigned by automated scoring systems. Powers, Burstein, Chodorow, Fowles, and Kukich (2001) solicited subject-matter experts to write essays that they thought would receive higher scores than they deserved from an automated essay scoring engine (and indirectly to provide evidence about the strategies they believed would lead to inflated scores). Powers, Cumming, and Kantor (2011) conducted a survey of individuals who had taken the Test of English as a Foreign Language (TOEFL®), in which they inquired about the test takers' beliefs and opinions regarding automated scoring technology generally, including

Derrick Higgins. Civis Analytics, 4875 N. Ashland Ave., Chicago, IL 60640; dhiggins@civisanalytics.com. Michael Heilman, Natural Language Processing and Speech Group, Educational Testing Service.

what response features they believed the technology might be most sensitive to. Powers et al. found that opinions on this subject varied widely, and were in many cases contradictory. Bejar (2013) pursued a new and important continuation of this line of research, by evaluating the effect that a gaming strategy based on lexical substitution—replacing common words in an essay with less common ones—has on an automated scoring engine. Studies such as McGee (2006) and Jones (2006) have reported on the experiences of instructors using automated scoring technology in a classroom context, including aspects of responses that these systems seem unable to appropriately assess. Such studies on the attitudes, beliefs, and experiences of instructors and examinees can inform research hypotheses about gaming strategies that are likely to be encountered in operational practice.

Another related area of research has to do with procedures for monitoring the performance of automated scoring systems over time, once they have been put to use in an operational environment. For example, Trapani, Bridgeman, & Breyer (2011) examined the causes of, and appropriate response to, separation between human and automated scores subsequent to the operational launch of automated CR scoring. Such monitoring evaluations may sometimes suggest the possibility of test-gaming strategies being applied. For example, Trapani et al. (2011) found greater discrepancies in mean scores assigned by the automated system relative to human raters across country groups, which could suggest differential application of gaming strategies, but could also be attributable to changes in human rating tendencies or other factors.

Other research on automated scoring technologies has established the need for strong alignment between the targeted construct and the methods used in scoring, in order to reduce the gap between theory and execution that could be exploited through construct-irrelevant strategies. Bennett and Bejar (1998) situated the issue of automated scoring in an evidence-centered design framework, and stressed the need for adequate construct representation within an automated scoring system, based on the interdependency of task design, construct definition, and the development of evidence models for tasks. Later research such as Quinlan et al. (2009), Xi, Higgins, Zechner, and Williamson (2012), Foltz, Streeter, Lochbaum, and Landauer (2013), and Schultz (2013) attempted to elucidate the internal structure of specific automated scoring engines, and indicated the degree of alignment with a predefined construct for the task or assessment.

There is also a fairly broad set of research studies concerned with the development of methods for filtering or flagging responses that are anomalous in some way (including many subclasses of anomalies that often correspond to "gaming" strategies). Higgins, Burstein, and Attali (2005) discussed methods for filtering anomalous essay responses based on vocabulary usage patterns, and these methods are further refined by Louis and Higgins (2010). Lochbaum, Rosenstein, Foltz, and Derr (2013) presented an overview of methods for detecting anomalous essays based on outlier identification. Yoon and Higgins (2011) and Cheng and Shen (2011) discussed methods for flagging anomalous spoken responses. Similarly, in the context of educational data mining, Baker, Corbett, Koedinger, and Wagner (2004) report on the development of a method to detect off-task student behavior intended to manipulate the automated tutoring system.

Finally, there is some research on related machine learning and/or natural language processing tasks that involve an "adversarial" aspect that introduces a risk that users may attempt to subvert the algorithm. Such tasks include spam filtering (cf. Sahami, Dumais, Heckerman, & Horvitz, 1998; Yih, McCann, & Kołcz, 2007), and the detection of malicious advertisements on the web (cf. Sculley et al., 2011). The approach to counteract gaming strategies taken in such contexts typically involves rapid retraining and redeployment of the algorithm to counter the strategies observed. In an assessment context, this approach is not always available or effective, because scoring behavior is expected to remain constant over time for psychometric reasons, and because the large scale of test administrations mean that a reactive approach may fail to address a gaming issue before it causes serious problems.

Measuring the Susceptibility of Automated Constructed-Response Scoring to Construct-Irrelevant Response Strategies

As discussed in the previous section, the issue of how to model the general susceptibility of automated scoring systems to construct-irrelevant response strategies is relatively unexplored, so previous research is a poor guide to how such a measure might be constructed. However, the properties that such a measure should have can be inferred based on the uses to which it would ideally be put.

First, a measure of gameability should be applicable to all CR tasks (or at least a very large proportion of them). Measures that are specific to a particular type of task (such as essays) in a particular type of test (e.g., writing placement) for a particular population (e.g., college freshmen) will be much less useful than those that can be applied to a wide variety of item types, content domains, and assessment types.

However, the measure should also be sensitive to the important differences that obtain between different types of CR tasks. Different types of items are designed to assess different skills, and so the types of evidence in student responses that automated scoring systems for these items are sensitive to will differ. In fact, sometimes superficially similar item types may be associated with very different scoring rubrics, and the methods needed for scoring them can differ substantially as well. As a consequence, gaming strategies that are effective for different types of items (or that students believe to be effective) may differ, and therefore any measure we develop to assess susceptibility to gaming must be flexible enough to account for the different types of strategies that could be applied.

A gameability measure needs to support comparisons between different automated scoring methods, whether they be different scoring engines (software implementations), different scoring models (statistical parameterizations of the same engine), or model variants (families of related models using the same engine).

Finally, an ideal measure of gameability would also support comparisons between different item types, complete tests, and testing programs. Essentially, the measure should be on the same scale across all of these different contexts of use.

Outline of Method Proposed

The proposed framework for evaluating the susceptibility of a scoring method to gaming strategies consists of four steps: hypothesis, simulation, optimization, and evaluation.

The *hypothesis* step involves identifying a set of candidate gaming strategies that are to be considered in connection with a particular task type. Candidate strategies may be identified based on research into test takers' perceptions of automated scoring (e.g., Powers et al., 2011), based on an understanding of the methods used by the automated scoring engine that may be fallible in some areas, or simply based on anecdotal evidence of response strategies previously observed for an assessment. By explicitly including a set of hypotheses about gaming strategies to be considered in the measure, we allow the framework itself to be completely general, while still ensuring that the measure takes into account the differences between task types.

The *simulation* step consists in operationalizing the gaming strategies hypothesized to be of interest, so that they can be emulated computationally (using natural language processing or other methods appropriate to the response type). The simplest way of defining a strategy as a computational operation will generally be to define a transformation function $s(\cdot)$ operating on a student response R, so that gaming behavior is essentially conceptualized as an augmentation of, or variation on, the type of response a student would otherwise provide. In some cases—where gaming strategies are quite extreme—it may make more sense to simulate hypothesized gaming strategies as response generation functions, rather than as transformations applied to existing responses. This approach can be subsumed under the current framework by defining the function $s(R)$ in such a way that its value does not depend on the properties of R. Each simulated gaming strategy may be applied to a greater or lesser degree, and we therefore assume that strategies are parameterized according to the aggressiveness with which they are used. The transformation function implementing hypothesized strategy m with parameter value n is represented as $s_m^n(\bullet)$. The composition of two such transformation functions is represented as $\left(s_m^n \circ s_q^p\right)(\bullet)$.

The next step in the proposed framework is to perform an *optimization* to find the best combination of strategies for improving test takers' scores. In order to perform an optimization, we require an objective function to optimize. From the perspective of the test taker, the goal is to achieve the highest score possible, so in this framework, the optimization is performed to maximize the score that is assigned to each response by the automated scoring system (or equivalently, the score increase relative to some baseline response behavior). The scoring function $Score(\cdot)$ may encompass other aspects of operational scoring beyond the artificial-intelligence score estimation techniques of the scoring engine, including methods for filtering anomalous responses and associated score imputation techniques, according to the policies of the operational programs for which the scoring model is to be used. Equation 1 introduces the "gameability metric" $\mathbf{\Gamma}$, and shows the optimization to be performed in order to identify the most effective joint gaming strategy (combination of parameter values for individual strategies) for a task.

$$\Gamma = \frac{1}{N}\max_{\{\vec{n},\vec{m}\}}\left\{\sum_{i=1\ldots N}\text{Score}\left[\left(s_{m_0}^{n_0}\circ s_{m_1}^{n_1}\circ\cdots\right)(R_i)\right]\right\} - \frac{1}{N}\sum_{i=1\ldots N}\text{Score}\left(R_i\right) \quad (1)$$

The optimization itself may be conducted in a number of ways. If the parameter space spanning all hypothesized gaming strategies is sufficiently small, it may be feasible to conduct an exhaustive search, evaluating the effectiveness of all joint strategies. However, the parameter space is exponential in the number of distinct gaming strategies considered, and polynomial in the number of parameter values allowed per strategy, and exhaustive search can therefore quickly become intractable. There is an extensive literature on methods for heuristic search through large parameter spaces, and known methods could be applied here, as well. For example, Hoos and Stützle (2005) provide a survey of "local search" methods that make successive, small changes to an initial parameterization in order to improve the value of an objective function (e.g., greedy search, beam search, and simulated annealing).

It should be noted that the assumption underlying this optimization step is that the susceptibility of an automated scoring system to gaming behavior is best represented by the effectiveness of the *most effective* gaming strategy. This is a plausible starting point, but it is not certain that construct-irrelevant response strategies observed in operational practice will ultimately converge toward this optimal strategy. Because test takers have limited opportunities to interact with the scoring system, often have limited information about it, and may have imperfect control over their actual execution of gaming strategies, the strategies they use in practice may be somewhat less effective. The measure of susceptibility developed here, therefore, may represent something like an upper bound on the effectiveness of practical strategies.[1]

Finally, the *evaluation* of the relative susceptibility of scoring methods (or items, or item types, or tests) is conducted simply by calculating the gameability metric Γ for each condition of interest. Because Γ itself is on the same scale as the score for an item, it has a relatively straightforward interpretation as the score increase expected using the most effective joint gaming strategy available. Comparisons between conditions can therefore be made on a fairly intuitive basis, although significance testing based on comparisons across multiple samples is feasible and advisable.

A Case Study: Short-Answer Scoring

The method described above for assessing the susceptibility of an automated scoring system to construct-irrelevant gaming strategies will be demonstrated through application to the task of automated short-answer scoring. Short-answer questions are a type of task for which questions of validity and potential test gaming strategies are particularly critical, given the consequential use for which automated scoring systems are being considered with respect to these tasks. A majority of the CR items to be included on end-of-year assessments developed by the Smarter Balanced consortium will be short-answer questions,[2] and if automated scoring systems are used for these tasks, there will be strong financial imperatives to use them as the sole scoring mechanism for most students. (While a hybrid model incorporating both human and automated scoring of each response is common for essay scoring, the costs of such a model would be harder for a program to support, given the much higher volume of short answers to be scored per student.)

The second phase of the Automated Student Assessment Prize (ASAP Challenge; cf. Shermis, 2013) was held during the summer of 2012, aiming to assess the state of the art in automated scoring technologies for short-answer questions. (The first phase, held in 2011, had focused on automated scoring of essays.) This ASAP Short Answer Scoring Challenge resulted in the development of a number of resources

Table 1. Teams Awarded Prizes in ASAP Short Answer Scoring Challenge

Team	Prize	Weighted Kappa
Luis Tandalla	First	.74717
Jure Zbontar	Second	.73892
Xavier Conort	Third	.73662
James Jesensky	Fourth	.73392
Paweł Jankiewicz & Stanley Peters	Fifth	.73094

that are well suited to support a study of susceptibility to gaming. The effort involved the public release of a large set of student responses to a variety of short-answer questions from state assessments, which could be used as a starting point for computational simulations of gaming strategies. The competition also resulted in the development and open-source release of scoring engines tailored to the tasks used for the contest. Because these engines were created independently, it is reasonable to think that they might differ in ways that affect their sensitivity to gaming. And because the software is freely available and documented, it is possible to attribute observed differences to particular design decisions.

The ASAP Short Answer Scoring Challenge

The ASAP Challenge was sponsored by the Hewlett Foundation, which provided a prize fund of $100,000 to encourage participation. It focused on the state of the art of automated short-answer scoring technology for the types of tasks likely to be included in new, multistate assessments then under development by the two major multistate organizations to receive federal funding under the Race to the Top Initiative: the Partnership for Assessment of Readiness for College and Careers (PARCC) and the Smarter Balanced consortium. Ten different short-answer questions were used for the contest, drawn from the domains of science and English language arts. Approximately 1,800 responses to each task were provided for system training, with another 600 or so used for periodic system evaluations (the "leaderboard"), and another approximately 600 responses used as a held-out sample on which to produce the final rankings of participating systems. Systems were evaluated based on how closely their predictions agreed with the holistic ratings provided by human graders for each response (on either a 0–2 or a 0–3 scale, by item), using the quadratic weighted kappa measure of agreement.

Many participants committed to releasing the code for their systems under an open-source license and to authoring a document describing their system's methodology (as these conditions were requirements for eligibility for cash prizes). Table 1 lists the five teams that ultimately were awarded prizes for the ASAP Short Answer Scoring Challenge, together with their final aggregate weighted kappa agreement measures on the test set. Three other teams achieved agreement scores high enough to have earned prizes, but chose not to release their engines' code under an open-source license. A team sponsored by Measurement Incorporated, incorporating contributions from independent data scientists, achieved the highest overall score of .74794.[3] Educational Testing Service also participated, obtaining a weighted kappa of .73498. Finally, a team comprised of Stefan Henß and Momchil Georgiev finished with an aggregate weighted kappa of .73495.

It is important to note that the scoring algorithms developed in the course of the ASAP Challenge were optimized to produce scores that agree with human ratings on the competition test data as highly as possible, and *not* primarily with the goal of operational scoring. For this reason, the developers of these systems could not be expected to ensure that their methods were invulnerable to gaming strategies that test takers might apply in an operational context. However, analysis of these systems is still quite relevant to the goal of evaluating the susceptibility of operational engines to gaming. For one thing, there is some possibility that these engines *will* nevertheless be used operationally: the ASAP Challenge was informed and supported by the major state assessment consortia, and at least one of the consortia is pursuing research on the effectiveness of open-source engines in scoring open-ended tasks (CTB-McGraw Hill, 2013). For another, the open-source engines provide a good starting point for all future research related to gameability, since they are freely available and allow for replication of reported results.

Open-Source Short Answer Scoring Engines

The scoring systems that were released under open-source licenses at the conclusion of the ASAP Challenge can be divided into two general categories based on the methods they employ. Most of the systems (those ranked 2–5 in Table 1) use relatively simple features such as the presence of particular words or word classes in a response, and incorporate this evidence into a complex multilevel statistical modeling architecture to assign a score. The only system that diverges significantly from this general framework is that of Luis Tandalla, which augments the core set of predictive features in the engine with features based on manually crafted answer patterns.

Because of issues related to the computing platform for which each engine was developed and dependencies on system libraries, only three of the systems listed in Table 1 could be made functional in the context of the experiment described below. These three systems are described here briefly before proceeding to the experimental setup.

Luis Tandalla (Tandalla, 2012). As indicated above, this system differs significantly from the others developed for the ASAP Challenge in that it attempts to identify particular linguistic patterns in responses that can be understood as typical answer types (rather than looking only at superficial features as direct predictors of the total score). These answer patterns are identified in Tandalla's system both through manually crafted rules and by means of statistical methods. Rule-based answer patterns are defined in terms of regular expressions, a language for specifying patterns over text strings in computer programming. Statistical pattern-identification models are developed by first manually labeling a set of responses instantiating the pattern, and then training a statistical classifier to discriminate between text segments that do and do not include the pattern. Both methods involve a great deal of manual effort per item. These answer patterns are combined with other evidence (such as the presence of particular words and word sequences) in a set of statistical predictors, whose outputs are averaged to produce the final score assignment. (As with all systems developed for the ASAP Challenge, the details of data preprocessing and other technical steps may be found in the technical methods paper hosted on the contest website.)

Jure Zbontar (Zbontar, 2012). Contrasted with the system of Luis Tandalla, which involved meticulous manual tailoring of the system to particular items, the system of Jure Zbontar is very simple. It applies a single, generic modeling structure to all items, although the models for scoring different items are parameterized differently. The model structure uses atomic features that are directly observable in responses, with little need for natural language processing analysis. Rather than using the words in a response as a cue to the score that should be assigned, this system's features are at an even lower level: sequences of characters within a word. In fact, six different variants of these "character n-gram" features are used, depending on how long a sequence is considered, whether any spelling correction is done, and whether the space is reduced using singular value decomposition. These character n-gram features are then input to an ensemble of statistical regression models, whose predictions are aggregated and rounded to yield the final score assignment.

Xavier Conort (Conort, 2012). Finally, the system developed by Xavier Conort uses a wide variety of automatically generated features in a stacked machine-learning prediction model. (A stacked model is one in which the predictions of multiple statistical prediction models are aggregated in some way by a "meta-classifier" to produce the final prediction.) Many of the system's features are *lexical*, meaning that they encode the presence of a single word or other highly local text element (such as a short sequence of words or word stems). This reliance on lexical features is a common characteristic of many ASAP Challenge engines, but the system developed by Xavier Conort does also include other features focused more on mechanical aspects of student responses, such as response length, the frequency of spelling errors and transition words, and the presence of particular types of punctuation.

Method

The method proposed here for evaluating the susceptibility of automated scoring systems to gaming strategies requires five components:

1) A specification of the *scoring method.*
2) A set of *hypotheses* about potential gaming strategies, and methods for computationally simulating the effects of such strategies.
3) A *search method* for exploring the space of possible atomic strategies in order to identify the optimal conjunction of strategies.
4) A *baseline* to which the effectiveness of gaming strategies can be compared.
5) An *evaluation metric* to be used in ranking conjoined gaming strategies.

In this case, the *scoring method* is fairly straightforward, as we are applying preexisting scoring engines calibrated for the ten ASAP short-answer tasks. In operational practice, however, automated scoring engines are typically augmented with methods for flagging anomalous responses, so that they can be singled out for special processing (such as review by human raters). In order to better simulate operational practice in this exploratory study, we also developed a simple filter to identify responses that are clearly uncharacteristic of normal response behavior. This filter flags as anomalous any response that is longer than the mean response length for a given task by at least 4 *SD*. The utility function (or scoring function) for a response R is thus given as

$$Score(R) = \begin{cases} \text{ASAP Scoring Model (R)} & \text{if Filter (R)} = 0 \\ 0 & \text{otherwise} \end{cases}$$

Responses using construct-irrelevant response strategies that are flagged by the filter will likely be evaluated by human raters and assigned a zero score; therefore, their utility to the test taker is zero.

For this experiment, we entertain three *hypotheses* about what gaming strategies test takers may employ in an attempt to fool the automated scoring system. These are motivated in part by previous research on students' perceptions of the biases of automated scoring (e.g., Powers et al., 2011), which indicates that some students believe computers to be influenced by writing mechanics, vocabulary selection, and response length. The strategies chosen are also informed by knowledge of the features used by the open-source scoring engines investigated. Since many of the algorithms use "bag of words" methods (models that rely on lexical features alone), it is natural to look at ways of manipulating the words used in a response without regard to higher-level linguistic structure (such as the order of words, the structure of phrases, or the meaning expressed by the response as a whole).

1. The first hypothesis is that the length of responses alone may influence the score assigned by the engine. In order to simulate the effect of padding responses without adding meaningful content, we examine the effect of concatenating multiple copies of the same response before submission to the scoring engine. Copies of the response may be appended zero, one, or two times to the original.
2. The second hypothesis is that the use of words from the question itself may increase the score assigned by the engine. We simulate the effect of trying to incorporate more vocabulary from the question itself by appending to each response a random selection of content words from the stimulus materials for the item (encompassing the question text as well as any associated reading passage or figure). The effect of adding 0, 5, 15, and 30 words to each response was investigated.
3. The third hypothesis is that the use of general academic words may increase the engine's score. We simulate the effect of trying to incorporate such academic vocabulary by appending to each response a random selection of words from the academic word list of Coxhead (2000). The effect of adding 0, 5, 15, and 30 words to each response was investigated.

Given this limited set of gaming strategies to be evaluated, and the limited number of parameterizations of each atomic strategy, it is possible to use exhaustive search as the *optimization method*. All possible conjunctions of atomic strategies are explored, and the one yielding the greatest utility (score increase) will be used to represent the system's maximum susceptibility to gaming.

The *baseline* against which gaming strategies were assessed was the set of scores obtained in actual operational practice. (For the purposes of this study, we make the simplifying assumption that these were free of construct-irrelevant gaming behavior. Because automated scoring was not used to

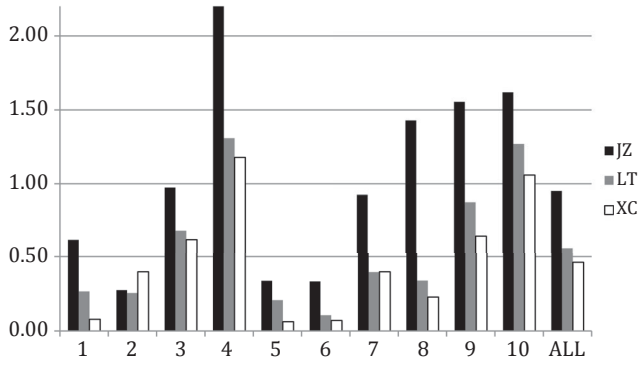

FIGURE 1. Standardized Γ statistic across all ten ASAP short answer scoring tasks, and, for the composite set of tasks, for three scoring engines.

score the actual state assessments, any gaming strategies that students applied would certainly not be intended to fool such systems; however, it must be acknowledged that students may have used strategies in this data set that were designed to fool *human* raters.) Gaming strategies are simulated as computational transformations of actually observed responses, so that the success of those strategies can be assessed by examining the change in score that this transformation produces. Responses already receiving the highest-possible score for a given task are excluded from the set to which gaming strategies are applied, as students who know the answer or have a strong understanding of the content area will have less to gain by responding in a construct-irrelevant fashion.[4]

Finally, the *evaluation metric* that was applied in this pilot study is simply the degree of increase in mean score obtained through the application of the optimal gaming strategy identified in the search space. This difference in mean score between the most effective joint gaming strategy and the baseline response behavior will be referred to as Γ_{ASAP}, and is an instantiation of the Γ family of metrics introduced above:

$$\Gamma_{\text{ASAP}} = \frac{1}{N} \max_{\{a,b,c\}} \left\{ \sum_{i=1\ldots N} \text{Score}\left[\left(s_1^a \circ s_2^b \circ s_3^c\right)(R_i)\right] \right\} - \frac{1}{N} \sum_{i=1\ldots N} \text{Score}\,(R_i)\,. \quad (2)$$

We also define $\Gamma_{\text{ASAP-std}}$ as Γ_{ASAP} normalized by the standard deviation of scores assigned to all responses to a given item. That is to say, $\Gamma_{\text{ASAP-std}}$ expresses the impact of a given set of gaming strategies in terms of the number of standard deviations of difference it yields in the ultimate score received, rather than directly in raw score points. Because the ten tasks in the ASAP Challenge did not all have the same score scale, we use $\Gamma_{\text{ASAP-std}}$ rather than Γ_{ASAP} when aggregating results across tasks.

Results

Figure 1 shows the overall results of calculating the $\Gamma_{\text{ASAP-std}}$ statistic for each ASAP short-answer scoring task, for the engines developed by Jure Zbontar (JZ), Luis Tandalla (LT), and Xavier Conort (XC). Results are also provided for the composite of all ten tasks (ALL). Note that the value presented for ALL tasks is not simply the mean of the values presented for each task individually, since different values of the gaming parameters may have resulted in the maximum success in inflating candidates' scores for each task. The value reported for all tasks, instead, reflects the maximum increase in score across the ten tasks that can be achieved using the *single* best joint gaming strategy applied consistently across all tasks.[5] These results are provided in tabular format in Tables 2–4, together with an indication of which parameter values resulted in the optimal performance for each task.

Figure 1 demonstrates that the susceptibility of different automated scoring engines varies substantially. The methods developed by Luis Tandalla (LT) and Xavier Conort (XC) show some susceptibility to gaming, with the optimal strategy yielding a mean increase of .56 and .46 *SD* in score, respectively, across tasks. The scoring engine of Jure Zbontar (JZ), however, is affected more dramatically by these simulated gaming strategies, showing an increase of almost one full standard deviation (.95) for the optimal such strategy.

Figure 1 further demonstrates that susceptibility to gaming varies a great deal across different types of short-answer questions. Because the questions in this data set were not selected specifically to support comparisons across subject areas and design parameters, there is a limit to how much can be inferred about the effect of different question characteristics on susceptibility to gaming; nevertheless, certain patterns do emerge. First, there seems to be a difference between science questions (1, 2, 5, 6) and English language arts questions (3, 4, 7, 8, 9, 10) with respect to gameability, with ELA questions quite a bit more susceptible to the strategies applied here. This may indicate that the scoring of science questions is less subjective, and tends to require that particular key concepts be referenced in order for credit to be awarded (so that adding more, tangentially relevant material is unlikely to improve the score given to a response). Another generalization suggested by the results in Figure 1 is that questions that draw on an outside body of knowledge (5, 6) tend to be less vulnerable to gaming than questions that are grounded in a reading passage or other longer stimulus (1, 2, 3, 4, 7, 8, 9, 10). This is intuitively plausible, as the stimulus passage provides a source from which vocabulary and other information can be drawn and incorporated into the response (either legitimately or as part of a construct-irrelevant strategy), whereas questions that draw on outside knowledge do not present test takers with access to domain-specific vocabulary. Finally, there is some limited evidence that designing tasks to elicit specific, localized information from a passage rather than information that may be dispersed across many locations within a passage may reduce the impact of gaming strategies. Tasks 3 and 4 are actually based on the same stimulus passage, with the former using it as the basis of a very targeted, specific question, and the latter a much more general one for which supporting evidence could be drawn from multiple places in the passage. As Figure 1 indicates, the scoring of task 3 was considerably less susceptible to the gaming strategies applied than the scoring of task 4.[6]

In addition to the assessment of these scoring engines' overall susceptibility to gaming strategies, it is important to understand which specific strategies are found to be most effective for each engine. Figures 2–4 present one measure to this determination: the change in score that is effected by

Table 2. Γ Statistics Associated With the JZ Scoring Engine for Each of the Ten ASAP Short-Answer Scoring Tasks, and All Tasks Combined, Together With the Gaming Parameters That Yielded This Maximum Level of Susceptibility

	JZ				
Task	Response Repeats	# Stimulus Words	# Academic Words	Γ_{ASAP} (Raw Score)	$\Gamma_{ASAP\text{-}std}$ (Standardized)
1	2	30	30	.65	.62
2	0	30	0	.27	.27
3	0	30	30	.65	.97
4	2	30	30	1.34	2.20
5	0	30	30	.20	.34
6	0	30	30	.22	.33
7	1	30	30	.76	.93
8	1	30	30	1.22	1.43
9	0	30	30	1.19	1.56
10	1	30	30	1.13	1.62
ALL	1	30	30		.95

Table 3. Γ Statistics Associated With the LT Scoring Engine for Each of the Ten ASAP Short-Answer Scoring Tasks, and All Tasks Combined, Together With the Gaming Parameters That Yielded This Maximum Level of Susceptibility

	LT				
Task	Response Repeats	# Stimulus Words	# Academic Words	Γ_{ASAP} (Raw Score)	$\Gamma_{ASAP\text{-}std}$ (Standardized)
1	2	30	30	.28	.27
2	1	30	30	.25	.26
3	2	30	30	.46	.68
4	2	30	5	.80	1.31
5	2	30	30	.12	.21
6	2	30	30	.07	.10
7	2	30	30	.32	.39
8	1	30	30	.29	.34
9	2	30	30	.67	.88
10	2	30	30	.89	1.27
ALL	2	30	30		.56

Table 4. Γ Statistics Associated With the XC Scoring Engine for Each of the Ten ASAP Short-Answer Scoring Tasks, and All Tasks Combined, Together With the Gaming Parameters That Yielded This Maximum Level of Susceptibility

	XC				
Task	Response Repeats	# Stimulus Words	# Academic Words	Γ_{ASAP} (Raw Score)	$\Gamma_{ASAP\text{-}std}$ (Standardized)
1	0	15	30	.08	.08
2	1	30	15	.39	.40
3	1	30	30	.42	.62
4	0	30	30	.72	1.18
5	1	15	30	.04	.06
6	1	0	30	.05	.07
7	0	30	30	.33	.40
8	1	30	15	.19	.23
9	2	30	30	.49	.65
10	1	30	30	.74	1.06
ALL	1	30	30		.46

applying only one of the three gaming strategies considered here, to varying degrees.

Figure 2 demonstrates that the LT engine is the most susceptible to gaming by means of simply padding out the response by copying. When two copies of the response are appended to the original, this results in a mean increase of .25 score standard deviations, across tasks. The effect on other engines is less than .1 *SD*. We can speculate that the LT engine is especially sensitive to repeated responses because it attempts to identify specific linguistic patterns as evidence of correct response behavior. The longer a (linguistically well-formed) response is, the greater the probability that a match will be found, and the higher the expected score.

Figure 3 shows that the JZ engine is the most susceptible to gaming by adding random words from the stimulus materials for the item. Adding 30 random content words results in a

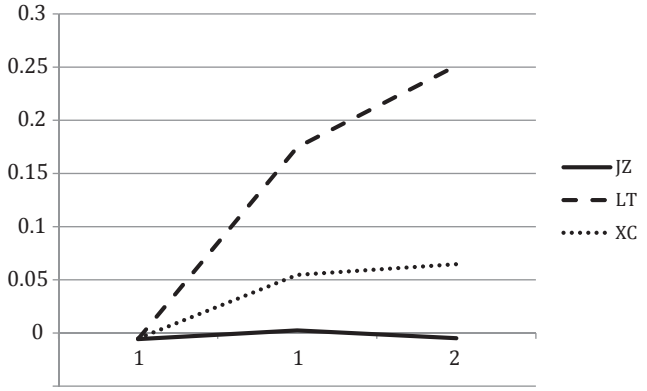

FIGURE 2. Standardized Γ statistic as a function of number of times response is repeated, applying no other gaming strategies.

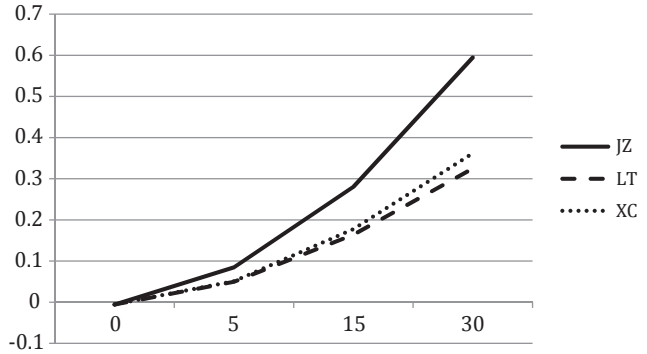

FIGURE 3. Standardized Γ statistic as a function of number of random words from item stimulus added, applying no other gaming strategies.

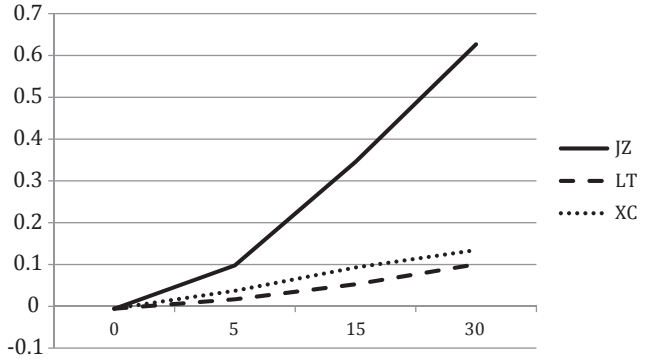

FIGURE 4. Standardized Γ statistic as a function of number of random academic words inserted, applying no other gaming strategies.

$\Gamma_{\text{ASAP-std}}$ increase of .59 for the JZ engine, but only around .35 for the other two engines. It is somewhat surprising that the XC and JZ engines behave so differently in response to this gaming strategy, given the similarities in their general architecture (using features related to the particular words and character sequences found in a response to predict the score). One difference that may explain this divergence, at least in part, is that the JZ engine relies *solely* on lexical features in determining the score to be assigned, while the XC engine also includes other features not solely dependent on the set of words and subword units in the response. (These include the frequency of spelling errors, discourse transition words, and punctuation elements.)

Finally, Figure 4 shows that the JZ engine is also the most susceptible to gaming by adding random academic words to a response (by a wide margin). Adding 30 random academic words results in a $\Gamma_{\text{ASAP-std}}$ increase of .63 for the JZ engine, but an increase of less than .15 for the other two engines. We again attribute this provisionally to the reliance of the JZ engine on lexical features only in its score assignment algorithm.

Discussion

This paper has demonstrated the need for an evaluation metric that can assess how susceptible automated scoring systems are to construct-irrelevant strategies for achieving inflated scores. This is an important characteristic of such engines to take into account when considering operational deployment, but one that is less straightforward to quantify than other characteristics such as reliability and fairness, due to its "counterfactual" nature. (We aim to reason not just about scoring behavior observed on previously collected responses, but about the behavior to be expected under future conditions that may differ from past experience. Therefore, we are faced with a lack of empirical data to support quantitative analysis.)

A preliminary framework has been presented for quantifying susceptibility of scoring engines to gaming strategies, as well. This framework is founded on the assumptions that we begin with strong hypotheses about what sorts of strategies might be employed in an attempt to fool the system, and that we can effectively simulate the effect of these strategies by using computational techniques for the generation or transformation of text. Where these assumptions are met, the evaluation framework proceeds by identifying the most effective set of strategies that can be employed, and taking that degree of score inflation as an index of susceptibility.

In our pilot analysis of scoring behavior of three of the prize-winning entries from the ASAP short-answer scoring competition, we found that simple gaming strategies showed varying degrees of effectiveness across the three engines, and across the ten short-answer questions in the data set. For particular combinations of engines and tasks, gaming strategies ranged from having almost no effect in some cases, to a mean increase of more than a full point in others.

One key lesson of this study is that simple gaming strategies can have non-negligible effects on the behavior of automated scoring engines. For this reason, it is crucial to assess the impact of likely strategies before operational implementation, to take steps to minimize the impact of such strategies in the scoring algorithm, and to develop response filters and other tools for identifying anomalous response types.

A second lesson is that the susceptibility of a scoring engine to gaming strategies is not always readily apparent based on consideration of the logical structure of the scoring engine. The model structures of the XC and JZ scoring engines evaluated here are much more similar to one another than either is to the LT engine, and one might therefore have assumed that they would show a similar degree of susceptibility to gaming, with the LT engine perhaps behaving differently. Instead, we

find that the JZ engine is considerably more susceptible to gaming, with the LT and XC engines clustering together with a similar, and lower, value of $\Gamma_{\text{ASAP-std}}$.

Future Work

Of course, this study represents only an initial step. It is to be hoped that this framework will inform future thinking about how to assess the impact of test gaming strategies on automated scoring, but there may well be better or more efficient methods of achieving that aim. In fact, the framework presented here is incomplete in many respects, and leaves many questions to be answered in future research.

Naturally, one important task for future research is to extend the application of this evaluation framework to engines that are in operational use for consequential assessments, or under consideration for operational introduction. Further experience with this evaluation framework will be needed in order to ensure that the results can inform operational practice appropriately.

Another area for extension is the application of the framework to a broader range of item types and classes of automated scoring models. While this paper has focused solely on automated scoring of short answers, similar issues of construct-irrelevant gaming behavior present a concern for essay tasks, spoken-response tasks, and indeed any open-ended task type that may be used in an assessment.

Future research to better understand the properties of gaming susceptibility measures derived under this framework is needed, as well. Understanding how sensitive these measures are to particular item types, populations, and characteristics of the score scale will be an important prerequisite to their use to inform operational practice.

The methods for simulation of gaming behavior described in this paper are only intended as examples of the kinds of simulation that can be done, and can certainly be improved upon. And of course, additional types of gaming behavior will need to be simulated in order to handle other sorts of scoring tasks, and likely even for adequate modeling of short-answer scoring. Incorporating such additional gaming strategies into this framework is straightforward, provided that computational techniques can be devised for simulating their effects. The optimization methods will need to be expanded as well, as not all hypothesis spaces will be restricted enough to allow for exhaustive search. In cases where a large number of atomic gaming strategies may be applied, where each atomic strategy has a large number of possible parameter settings that govern how it is applied, or where there are inter-dependencies between atomic strategies, alternate optimization techniques may need to be used.

It is also important that future research seek to validate the measures of susceptibility to gaming introduced here by using external criteria. We might hope, for example, that the measure of susceptibility to gaming that is calculated in advance of operational introduction of a scoring model would be predictive of the frequency of observed gaming behavior once the test is actually released. The stronger the link between the predicted effectiveness and observed incidence of test gaming behavior, the more useful this tool will be for operational testing. Such data would be difficult to obtain and present, both due to their sensitivity and due to the likelihood that any gaming behavior would be infrequent even when test takers suspect it to be effective. However, if data on actual student gaming behavior can be obtained, it will constitute the best possible empirical basis for making informed program decisions, and it is therefore an important goal to pursue.

Finally, we anticipate that research on how to safeguard the validity of automated scoring systems by detecting and flagging anomalous responses (which may reflect the application of gaming strategies) will continue, and may be informed by the measures of susceptibility described here. The susceptibility of algorithms to particular strategies may influence the prioritization of topics within this research domain, and also suggest new ways of insulating engines from the effects of gaming.[7]

Open Issues

In addition to these areas for technical refinement or elaboration, there are also conceptual issues related to evaluations of test-gaming behavior that will not admit of resolution in the foreseeable future.

First, there is not a clear dividing line between strategies that are truly construct-irrelevant, and can be labeled as "cheating," and those that are simply "test-wise" and constitute informed strategies for optimally deploying construct-relevant knowledge in the context of an assessment. To take a simple example, automated essay scoring systems are sometimes thought to reward the length of responses disproportionately (Powers et al., 2011). Advisors who counsel students to write longer essay responses may indeed intend this as a strategy for gaming the scoring engine, but they also may have the more innocuous intent of encouraging students to fully develop their arguments, rather than doing the bare minimum to satisfy the directions provided for the task. In the context of the framework proposed here for assessing the susceptibility of automated systems to gaming behavior, this means that decisions about what hypothesized gaming strategies to simulate will always be somewhat subjective and controversial.

The other difficult conceptual issue is that the application of test-gaming strategies is not restricted to contexts in which CR items are scored by computer. Human raters may also be susceptible to particular construct-irrelevant strategies. See Powers (2005) for a discussion of the importance of response length in determining human ratings, and the degree to which length may or may not encode construct-relevant information. Ideally, we could use the framework described here to provide a parallel evaluation of human raters' susceptibility to gaming; unfortunately, this presents a number of problems which do not admit of an easy solution. One challenge is that the gaming simulation strategies devised for this study are fairly simplistic, and produce responses that would be jarring to human eyes. (For instance, a list of academic vocabulary appended to a response would be sure to attract raters' attention, and would likely lead them to call in scoring leaders or otherwise flag the response.) Another challenge is that it is generally infeasible from a resource perspective to ask human raters to independently score the tens or hundreds of thousands of (simulated) CRs that this framework requires. Nevertheless, this does not mean that the targeted gaming strategies might not influence human ratings to an extent similar to any observed susceptibility of automated systems. It may be the case that other sorts of manipulations can be applied on a smaller scale to assess the susceptibility of human raters to gaming strategies. (See, for example, the grouping of

essays into rating batches by length by Attali, Lewis, & Steier, 2013.)

One future development is certain: that as automated CR scoring comes into operational practice across a broader range of assessments, strategies for gaming the technology will proliferate and be refined. The research community needs to develop tools that will allow us to assess these threats before operational release, as well as tools for monitoring and remediation.

Notes

[1]As noted by an anonymous reviewer, summative assessment systems may have parallel systems in use for formative purposes. If the same automated scoring models are used in both systems, students will have the opportunity to evaluate the impact of potential gaming strategies using the formative system, and may be better able to optimize their strategies for the summative test.

[2]Cf. sample Smarter Balanced tasks hosted at http://www.smarterbalanced.org/sample-items-and-performance-tasks/.

[3]For consistency with results reported by the organizers of the ASAP Challenge, results are reported to five decimal places. However, no statistical significance tests were reported for these system comparisons, and it is unlikely that the reliability of these agreement estimates justifies this level of precision.

[4]Different modeling decisions could be made with regard to the differential application of gaming strategies across subpopulations. While it is plausible that students with higher content mastery would be less likely to apply test gaming strategies, we are unaware of data that would allow us to quantify the degree to which this disparity exists. One alternative approach would be to simulate gaming behavior for all responses (including those receiving the highest score), with the likely effect that the gaming susceptibility metric would be reduced somewhat (because of the ceiling effect, and because the effect of gaming might be to reduce the score assigned, in some rare cases). Another alternative would be to simulate gaming strategies for only those candidates with the lowest level of mastery; this would likely yield some increase in the overall susceptibility metric. Of course, the most important consideration is that the same modeling parameters are applied consistently across all conditions to be compared (engines, tasks, gaming strategies, etc.).

[5]More formally, the values reported in Figure 1 represent

$$\frac{1}{N}\max_{\{a,b,c\}}\sum_{task=1\ldots10}\left\{\sum_{i=1\ldots N}\text{Score}_{\text{task}}\left[\left(s_1^a\circ s_2^b\circ s_3^c\right)(R_i)\right]-\sum_{i=1\ldots N}\text{Score}_{\text{task}}(R_i)\right\}$$

rather than

$$\frac{1}{N}\sum_{\text{task}=1\ldots10}\max_{\{a,b,c\}}\left\{\sum_{i=1\ldots N}\text{Score}_{\text{task}}\left[\left(s_1^a\circ s_2^b\circ s_3^c\right)(R_i)\right]-\sum_{i=1\ldots N}\text{Score}_{\text{task}}(R_i)\right\}$$

[6]The tasks themselves, as well as the student responses used for the ASAP Short Answer Scoring Challenge, are available at https://www.kaggle.com/c/asap-sas/data.

[7]One way in which this work could be applied to support more valid scoring and mitigate the effects of gaming, suggested independently by Shayne Miel and an anonymous reviewer, would be to feed responses that reflect gaming (real or simulated) back into the engine training data, so that the engine can learn to identify such strategies and handle them appropriately.

References

Attali, Y., Bridgeman, B., & Trapani, C. (2010). Performance of a generic approach in automated essay scoring. *Journal of Technology, Learning, and Assessment*, *10*(3), 1–17.

Attali, Y., Lewis, W., & Steier, M. (2013). Scoring with the computer: Alternative procedures for improving the reliability of holistic essay scoring. *Language Testing*, *30*(1), 125–141.

Baker, R., Corbett, A., Koedinger, K., & Wagner, A. (2004, April). *Off-task behavior in the Cognitive Tutor classroom: When students "game the system."* Paper presented at the ACM Workshop on Computer-Human Interaction, Vienna, Austria.

Bejar, I. (2013, April). *Gaming a scoring engine: Lexical and discourse-level construct-irrelevant response strategies in the assessment of writing*. Paper presented at the meeting of the National Council on Measurement in Education, San Francisco, CA.

Bennett, R., & Bejar, I. (1998). Validity and automated scoring: It's not only the scoring. *Educational Measurement: Issues and Practice*, *17*(4), 9–17.

Bernstein, J., Van Moere, A., & Cheng, J. (2010). Validating automated speaking tests. *Language Testing*, *27*, 355–377.

Bridgeman, B., Trapani, C., & Attali, Y. (2009, April). *Considering fairness and validity in evaluating automated scoring*. Paper presented at the meeting of the National Council on Measurement in Education, San Diego, CA.

Cheng, J., & Shen, J. (2011, August). Off-topic detection in automated speech assessment applications. Paper presented at the Interspeech conference, Florence, Italy.

Conort, X. (2012). *Short answer scoring: Explanation of "Gxav" solution*. ASAP Short Answer Scoring Competition System Description. Retrieved July 28, 2014, from http://kaggle.com/c/asap-sas/details/winners

Coxhead, A. (2000). A new academic word list. *TESOL Quarterly*, *34*, 213–238.

CTB McGraw-Hill. (2013, July). *Item development management for the field test and scoring management for the pilot test and field test*. Proposal submitted to the Smarter Balanced Assessment Consortium. http://www.k12.wa.us/RFP/pubdocs/SBAC16-17/CTBSBAC16-17Proposal.pdf.

Foltz, P., Streeter, L., Lochbaum, K., & Landauer, T. (2013). Implementation and applications of the Intelligent Essay Assessor. In M. Shermis & J. Burstein (Eds.), *Handbook of automatted essay evaluation* (pp. 68–88). New York: Routledge.

Higgins, D., Burstein, J., & Attali, Y. (2005). Identifying off-topic student essays without topic-specific training data. *Natural Language Engineering 12*(2), 145–159.

Hoos, H., & Stützle, T. (2005). *Stochastic local search: Foundations and applications*. San Francisco, CA: Morgan Kaufmann/Elsevier.

Jones, E. (2006). ACCUPLACER's scoring technology: When reliability does not equal validity. In P. Ericsson & R. Haswell (Eds.), *Machine scoring of student essays: Truth and consequences* (pp. 93–113). Logan: Utah State University Press.

Landauer, T. K., Laham, R. D., & Foltz, P. W. (2003). Automated scoring and annotation of essays with the Intelligent Essay Assessor. *Assessment in Education*, *10*(3), 295–308.

Lochbaum, K., Rosenstein, M., Foltz, P., & Derr, M. (2013, April). *Detection of gaming in automated scoring of essays with the IEA*. Paper presented at the meeting of the National Council on Measurement in Education, San Francisco, CA.

Louis, A., & Higgins, D. (2010, June). *Unsupervised prompt expansion for off-topic essay detection*. Paper presented at the Workshop on Building Educational Applications, Los Angeles, CA.

McGee, T. (2006). Taking a spin on the Intelligent Essay Assessor. In P. Ericsson & R. Haswell (Eds.), *Machine scoring of student essays: Truth and consequences* (pp. 79–92). Logan: Utah State University Press.

Page, E. B. (1968). The use of the computer in analyzing student essays. *International Review of Education*, *14*(2), 210–225.

Page, E. B. (1994). Computer grading of student prose, using modern concepts and software. *Journal of Experimental Education*, *62*, 127–142.

Powers, D. (2005). *Wordiness: A selective review of its influence, and suggestions for investigating its relevance in tests requiring extended written responses* (Research Memorandum No. 04–08). Princeton, NJ: Educational Testing Service.

Powers, D., Burstein, J., Chodorow, M., Fowles, M., & Kukich, K. (2001). *Stumping e-rater: challenging the validity of automated essay scoring* (Research Report No. 01–03). Princeton, NJ: Educational Testing Service.

Powers, D., Cumming, A., & Kantor, R. (2011). *Scoring the TOEFL® independent essay automatically: Reactions of test takers and test score users* (Research Memorandum No. 11–34). Princeton, NJ: Educational Testing Service.

Quinlan, T., Higgins, D., & Wolff, S. (2009). *Evaluating the construct coverage of the e-rater scoring engine* (Research Report No. 09–03). Princeton, NJ: Educational Testing Service.

Sahami, M., Dumais, S., Heckerman, D., & Horvitz, E. (1998). A Bayesian approach to filtering junk email. In *Proceedings of AAAI Workshop on Learning for Text Categorization* (pp. 55–62). Madison, WI: Association for the Advancement of Artificial Intelligence.

Schultz, M. (2013). The IntelliMetric™ automated essay scoring engine: A review and an application to Chinese essay scoring. In M. Shermis & J. Burstein (Eds.), *Handbook of automated essay evaluation* (pp. 89–98). New York: Routledge.

Sculley, D., Otey, M., Pohl, M., Spitznagel, B., Hainsworth, J., & Zhou, Y. (2011). Detecting adversarial advertisements in the wild. In *Proceedings of KDD 2011* (pp. 274–282). San Francisco, CA: Association for Computing Machinery.

Shermis, M. D. (2013, April). *Contrasting state-of-the-art in the machine scoring of short-form constructed responses*. Paper presented at the meeting of the National Council on Measurement in Education, San Francisco, CA.

Tandalla, L. (2012). *Scoring short answer essays*. ASAP Short Answer Scoring Competition System Description. Retrieved July 28, 2014, from http://kaggle.com/c/asap-sas/details/winners

Trapani, C., Bridgeman, B., & Breyer, F. J. (2011, April). *Using automated scoring as a trend score: The implications of score separation over time*. Paper presented at the meeting of the National Council on Measurement in Education, New Orleans, LA.

Williamson, D., Xi, X., & Breyer, F. J. (2012). A framework for evaluation and use of automated scoring. *Educational Measurement: Issues and Practice 31*(1), 2–13.

Xi, X., Higgins, D., Zechner, K., & Williamson, D. (2012). A comparison of two scoring methods for an automated speech scoring system. *Language Testing*, *29*, 371–394.

Yih, W.-T., McCann, R., & Kołcz, A. (2007, August). *Improving spam filtering by detecting gray mail*. Paper presented at the Fourth Conference on Email and AntiSpam, Mountain View, CA.

Yoon, S.-Y., & Higgins, D. (2011, June). *Non-English response detection method for automated proficiency scoring system*. Paper presented at the Workshop on Building Educational Applications, Portland, OR.

Zbontar, J. (2012). *Short answer scoring by stacking*. ASAP Short Answer Scoring Competition System Description. Retrieved July 28, 2014, from http://kaggle.com/c/asap-sas/details/winners

Zechner, K., Higgins, D., Xi, X., & Williamson, D. (2009). Automatic scoring of non-native spontaneous speech in tests of spoken English. *Speech Communication*, *51*, 883–895.