# Formative Essay Feedback Using Predictive Scoring Models

Bronwyn Woods, David Adamson, Shayne Miel, Elijah Mayfield
Turnitin
2020 Smallman St
Pittsburgh, PA 15222
[bwoods,dadamson,smiel,elijah]@turnitin.com

## ABSTRACT

A major component of secondary education is learning to write effectively, a skill which is bolstered by repeated practice with formative guidance. However, providing focused feedback to every student on multiple drafts of each essay throughout the school year is a challenge for even the most dedicated of teachers. This paper first establishes a new ordinal essay scoring model and its state of the art performance compared to recent results in the Automated Essay Scoring field. Extending this model, we describe a method for using prediction on realistic essay variants to give rubric-specific formative feedback to writers. This method is used in Revision Assistant, a deployed data-driven educational product that provides immediate, rubric-specific, sentence-level feedback to students to supplement teacher guidance. We present initial evaluations of this feedback generation, both offline and in deployment.

## 1 INTRODUCTION

Writing, though central to education, is labor intensive to assess. Teachers must balance giving students iterative practice and feedback with their own available time. As a result, Automated Writing Evaluation (AWE) tools have been a focus in industry and research for decades. Much of the existing work has been directed toward scoring over feedback, often for efficiency in high volume scoring applications such as testing. Nevertheless, there has been a persistent recognition that providing formative feedback is an essential goal for comprehensive automated tools [10, 23, 27].

We present the modeling framework that powers Revision Assistant, an educational tool for providing sentence-level, rubric-specific, formative feedback to students during the revision process. As shown in figure 1, Revision Assistant provides students with predicted scores on multiple rubric traits and highlights several sentences with trait specific suggestions or encouragement.

Though essay scoring is a well defined supervised learning task, there is no ground truth for feedback. There are many plausibly useful sets of feedback that could be given on a particular essay and no established evaluation criteria. We therefore frame the feedback problem as one of prediction on realistic essay variants, to identify regions in which revisions may have an impact on predicted scores. We find that a regularized ordinal logistic regression scoring model meets the demands of this extension of prediction to feedback, while still maintaining state of the art performance on scoring.

In Section 2 we introduce the problem of Automated Essay Scoring (AES) and position our approach among others used in academia and industry. Section 3 describes the way in which we use these models to select sentences for feedback, grounding this approach in pedagogical theory and practice. We then discuss how the standard assumptions and evaluation methods of predictive modeling shift for feedback generation. Section 4 outlines the deployment of our scoring and feedback system to hundreds of thousands of students. We conclude with a discussion of next steps and open questions in this area.

## 2 AUTOMATED ESSAY SCORING

Scoring models are featured explicitly in Revision Assistant, and also provide the foundation for feedback. Here we describe the task of supervised prediction of essay scores with respect to a given prompt and rubric.

### 2.1 Problem definition and existing work

In 1966 Ellis Page developed Project Essay Grade (PEG), kicking off 50 years of research into AES [21]. Methodology and computational power have advanced substantially, but the task has remained largely consistent. In traditional non-automated, large-scale essay scoring, two trained experts typically score each essay. A third expert resolves disagreements between them. The task of an AES system is to use scores from this traditional scoring process to train models that can score new essays as reliably as any individual rater.

The AES field grew out of a need to replicate the work of expert essay readers, and many approaches try to directly incorporate insights from those experts. Many systems, including Ellis's PEG, ETS's e-rater, and more, focus on feature engineering. They create a small to moderate number of expert-designed features meant to represent high level characteristics of writing [6, 7, 19, 21]. These may include measures such as coherence or lexical sophistication [32]. The connection between the constructs used by human experts and the AES system is generally emphasized as a central feature.

In contrast, an increasing body of work attempts to avoid laborious feature engineering by using large numbers of low-level textual features [22] or neural network derived word or paragraph embeddings [2, 29]. These systems use high dimensional modeling techniques, and relax the constraint that model features should mimic human reasoning. We use this approach, demonstrating with our feedback system that expert derived features are not required for interpretable output. Our results in this area are parallel


*KDD'17, , August 13–17, 2017, Halifax, NS, Canada.*

ACM ISBN 978-1-4503-4887-4/17/08…$15.00
https://doi.org/10.1145/3097983.3098160

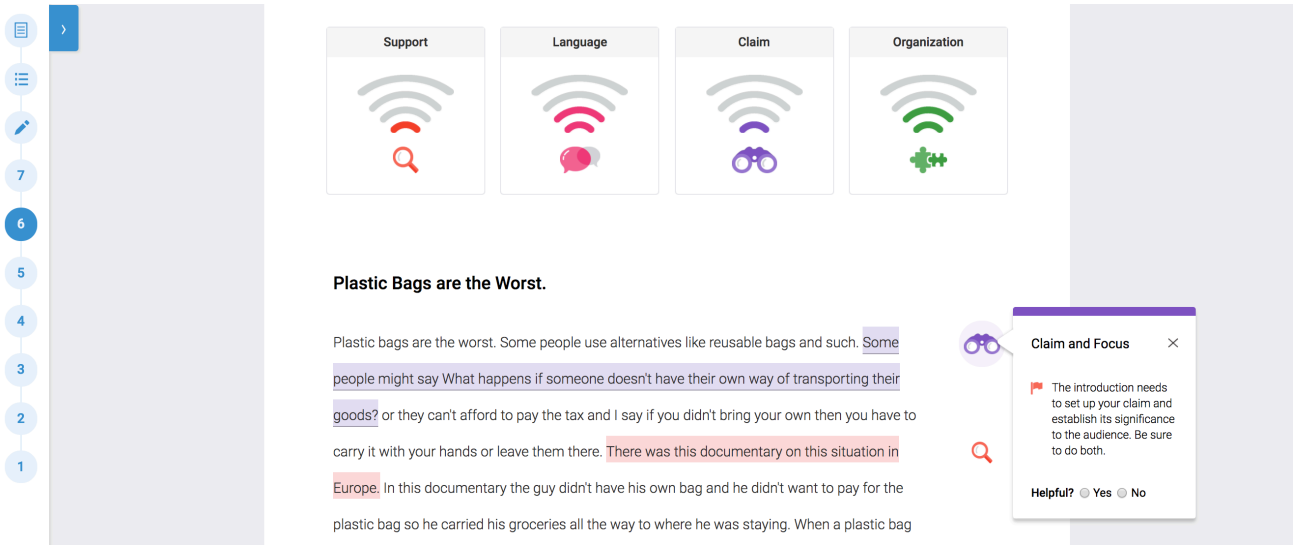


**Figure 1: An example of Revision Assistant in use. The signal bars at the top of the draft indicate predicted scores on four rubric traits. Highlighted sentences give targeted feedback to the student on strengths and weaknesses tied to the traits.**

to recent work in the deep learning domain on creating textual rationales for network predictions [16].

Academic AES research has been dominated for the last five years by the dataset from the 2012 Automated Student Assessment Prize (ASAP) competition [1]. This competition used essays written to eight prompts, scored on a variety of scales. The competition had a private phase with companies as competitors, followed by a public Kaggle competition with anonymized data. [26] provides a summary of the competition, and most recent research papers report their results using the same public dataset [2, 9, 11, 22, 29].

Essay scoring is commonly framed as a regression problem, despite the fact that scoring ranges are bounded integer scales [2, 7, 11, 21, 22, 29]. This approach is powerful when the score range is large, better approximating a continuous response, as is the case in several of the ASAP datasets. However, many rubrics, including those used in Revision Assistant, use a much smaller scale where regression may not be appropriate. Other work frames essay scoring as a classification problem [25]. This is perhaps better suited to a response variable with a small number of discrete scores, but loses any acknowledgement of the ordering of the scoring scale. We introduce here an AES system using ordinal logistic regression, accurately modeling the essay scores as both ordered and discrete.

## 2.2 Data and hand-scoring

We use two datasets in this paper. The first is the ASAP competition public data. Details of this dataset are available on the competition website, and in [26]. We use this data to compare our algorithms to recent literature using the same task. We omit the eighth ASAP dataset due to its very large scoring range, which is poorly suited to our modeling approaches. Table 2 includes summary statistics of the ASAP data.

The second dataset is a sample of 9 prompts available in Revision Assistant. This dataset includes double-scored training essays, as well as usage data from the deployed system. The prompts cover a range of genres and grade bands, with most essays being between 200 and 700 words. Table 3 provides summary statistics for this dataset. A list of the prompts and rubrics currently used in Revision Assistant is available in the online documentation [2].

A professional scoring vendor evaluated between 280 and 1000 essays per Revision Assistant prompt. Each essay received scores from two expert raters, with disagreements resolved by a third. The scores ranged from 1 to 4 for each trait, and all of these prompts were scored on four rubric-specific writing traits. Examples of rubric traits include *Use of Evidence*, *Clarity*, and *Organization*. Table 1 shows agreement measures for the expert raters on the Revision Assistant datasets. Agreement varies, but is generally between 0.6 and 0.8 Quadratic Weighted Kappa (see section 2.6). Note that around 10 raters worked on each prompt, so the measurement of agreement is between two arbitrary raters, not any specific individuals.

In general, intermediate scores of 2 and 3 are far more common than the extreme scores of 1 or 4. In addition, the resolution process tends to pull scores toward the center. This is a common pattern in our dual-rater scoring, but often results in the extreme scores being underrepresented and training sets being very imbalanced. To combat this in practice, we include essays in our training sets with the extreme scores if at least one rater scored them as such. When incorporating a new prompt into Revision Assistant, we evaluate the models with and without this process, against both original and modified scores. We work with the modified training sets in this paper to match what Revision Assistant uses in practice.

[1] https://www.kaggle.com/c/asap-aes

[2] https://guides.turnitin.com/Revision_Assistant/Prompt_Library

| | Trait QWKs | | | | Score Types | | Score Dist (%) | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | T1 | T2 | T3 | T4 | Res | Ext | 1 | 2 | 3 | 4 |
| P1 | 0.62 | 0.62 | 0.63 | 0.59 | 457 | 0 | 3 | 37 | 52 | 8 |
| P2 | 0.62 | 0.71 | 0.71 | 0.72 | 945 | 98 | 7 | 61 | 25 | 7 |
| P3 | 0.71 | 0.70 | 0.68 | 0.71 | 359 | 40 | 6 | 39 | 44 | 12 |
| P4 | 0.70 | 0.69 | 0.64 | 0.62 | 706 | 48 | 6 | 41 | 45 | 7 |
| P5 | 0.74 | 0.74 | 0.74 | 0.71 | 578 | 80 | 28 | 43 | 25 | 5 |
| P6 | 0.60 | 0.68 | 0.67 | 0.61 | 490 | 0 | 5 | 43 | 43 | 9 |
| P7 | 0.72 | 0.60 | 0.71 | 0.59 | 216 | 64 | 14 | 40 | 31 | 15 |
| P8 | 0.79 | 0.79 | 0.78 | 0.78 | 264 | 61 | 13 | 29 | 30 | 29 |
| P9 | 0.67 | 0.67 | 0.62 | 0.62 | 643 | 126 | 6 | 40 | 41 | 12 |

**Table 1: Hand scoring QWKs indicate the agreement between the original raters. Score Types indicate how many essays used the Resolved or Extreme scores in our training sets. Score Dist gives the percentage of essays with each score.**

## 2.3 Feature space

We take the approach of using large numbers of low-level textual features to represent essays. These features are simple to compute and easy to use in a variety of models. Included in the feature space are counts and binary occurrence indicators of word unigrams, bigrams and trigrams, character four-grams, and part of speech (POS) bigrams and trigrams. We use the Stanford part of speech tagger to generate POS features [30]. This feature space is generally quite large, with several orders of magnitude more features than scored essays. All of our models first remove extremely rare features and then use some form of model regularization to fit models in the face of large feature spaces.

In models with joint parameter estimation, we also include essay length as a predictor. It is generally the case that essay scores are strongly correlated with length. This is not surprising, as evaluation criteria such as inclusion of evidence and examples are difficult to achieve in shorter texts. Many AES systems choose not to include length as a predictor, as it is not *in and of itself* something that experts wish to evaluate. However, so many common features in models (such as counts of common words or parts of speech sequences) are proxies for length that its exclusion from a model is largely symbolic. We will argue in Section 3.3 that including and isolating essay length as an explicit model parameter strengthens feedback.

In addition to the scored essays for each prompt, we also use an unscored collection of thousands of essays collected from students using Revision Assistant in classrooms. This extra unsupervised data is used to inform rare feature removal, and to provide the basis for our off-topic essay detection mentioned in section 4. Future work in semi-supervised modeling may be able to make even more use of this data.

## 2.4 Baseline models

We present two simple models as baselines on the Revision Assistant data: a length-only classifier, and Naive Bayes using a comparable feature set to our proposed ordinal model. As demonstrated in table 2, our Naive Bayes baseline is already competitive with results from other researchers on the public datasets we evaluate.

*2.4.1 Length Only (LEN).* This baseline addresses the correlation between essay length and score. The model is a logistic regression classifier with a single predictor, namely the character length of the essay. This model has widely varying performance. In a few instances, the expert raters' scores are so closely correlated with length that length on its own is as effective as more sophisticated models. However, in the vast majority of cases, the length-only model under-performs other models.

*2.4.2 Naive Bayes (NB).* Our second baseline is a Naive Bayes model. The model uses binary features indicating the presence of word, character, and part of speech n-grams. We remove features that occur in fewer than 10 training set essays, and then use a cross-validated grid search to select a subset of this feature space, selecting 1000 to 5000 features using $\chi^2$ feature selection. This baseline performs well and is quick to train. However, the independence assumption that makes it efficient also leads to poor probabilistic calibration which makes it less suited for generating feedback, as we will discuss in section 4.2.

## 2.5 Ordinal logistic regression (OLR)

Though most AES systems, including our baselines, treat score prediction as a classification or linear regression problem, the response variable is actually ordinal. Because of the small, discrete, range of possible scores, a linear regression approach will suffer from violations of modeling assumptions. A classification approach treats each score as an independent category, losing ordering information.

Ordinal logistic regression (OLR) accounts both for the categorical nature of the scores and their ordering. We use the continuation odds ratio ordinal logistic regression model [1]. This model estimates the conditional probabilities of an essay receiving a particular score given that it scores *at least* that high. For an essay $Y$ and score $k$, $\mathbf{x}$ is the feature vector of essay $Y$, and

$$logit\left(P(Y = k|Y \geq k)\right) = \alpha_k + \beta^T \mathbf{x}. \tag{1}$$

Since the parameter vector $\beta$ does not depend on $k$, the model in equation 1 makes a proportional odds assumption. Though there is a separate intercept term $\alpha_k$ for each score point, the impact of the essay features on the odds of crossing each threshold is the same. As in [8], it is possible to relax this assumption by estimating separate parameter vectors $\beta_k$ for some $k$. We do so for the thresholds at the ends of the scoring scale, such that our model is

$$logit\left(P(Y = k|Y \geq k)\right) = \alpha_k + \beta^T \mathbf{x} + \left[\beta_{t1}^T \mathbf{x} \cdot I_{k=1}\right] + \left[\beta_{t3}^T \mathbf{x} \cdot I_{k=3}\right] \tag{2}$$

The features we use in the OLR model include essay length, counts of word, character, and POS n-grams, and binary indicators of the same n-grams. This is a very large feature space, requiring regularization to fit the logistic model. As described in [4, 8], the likelihood for this model can be factored into $K - 1$ independent likelihoods estimated simultaneously using an expanded $\mathbf{X}$ matrix. Because of the large feature space, we use Elastic Net regularization, with the regularization parameters chosen through a grid search.

## 2.6 Model performance

The Revision Assistant dataset is constrained by privacy requirements and cannot be made public. As validation that our score

predictions are competitive with current research, we tested our scoring algorithms on the publicly available ASAP dataset. Table 2 shows these results, compared against recently published results (the neural network approach of [29], and the Bayesian linear ridge regression and SVM approaches of [22], and the Timed Aggregate Perceptron approach of [9]). We see from these results that our OLR classification approach suffers when grading essays on large scoring ranges (ASAP 1, 7), but performs as the state of the art on prompts more similar to the Revision Assistant use case.

Scores for all evaluations are reported as Quadratic Weighted Kappa (QWK), a standard measurement in the AES field.

$$QWK = 1 - \frac{\sum_{i=1}^{K}\sum_{j=1}^{K} w_{ij}\mathbf{o}_{ij}}{\sum_{i=1}^{K}\sum_{j=1}^{K} w_{ij}\mathbf{e}_{ij}} \quad (3)$$

where $\mathbf{o}$ is the observed confusion matrix between true and predicted scores, and $\mathbf{e}$ is the expected matrix given random draws from the empirical distributions of predicted and true scores. The weight matrix $w$ is given by $w_{ij} = (i-j)^2/(K-1)^2$.

| Set | Essays | Scores | SVM | BLRR | TAP | NN | NB | OLR |
|---|---|---|---|---|---|---|---|---|
| 1 | 1783 | 2-12 | 0.78 | 0.76 | **0.82** | **0.82** | 0.79 | 0.77 |
| 2 | 1800 | 1-6 | 0.62 | 0.61 | 0.67 | 0.69 | **0.71** | **0.71** |
| 3 | 1726 | 0-3 | 0.63 | 0.62 | 0.64 | 0.69 | **0.71** | 0.70 |
| 4 | 1772 | 0-3 | 0.75 | 0.74 | 0.79 | **0.81** | 0.79 | **0.81** |
| 5 | 1805 | 0-4 | 0.78 | 0.78 | 0.78 | 0.81 | 0.79 | **0.82** |
| 6 | 1800 | 0-4 | 0.77 | 0.78 | 0.77 | 0.82 | 0.78 | **0.83** |
| 7 | 1569 | 0-30 | 0.73 | 0.73 | 0.73 | **0.81** | 0.78 | 0.63 |

**Table 2: QWK performance on the public ASAP prompts show that our OLR model achieves state of the art predictive performance. *SVM* and *BLRR* are from [22], *TAP* is the approach of [9], and *NN* is the CNN+LSTM approach in [29]. NB and OLR are our baseline (Naive Bayes) and ordinal models. For consistency with cited papers, we evaluated all models on the ASAP data using 5-fold CV, though the folds themselves are not identical to the referenced works.**

Table 3 shows the performance of our baseline and OLR models on the 9 Revision Assistant datasets, evaluated in 10-fold cross validation with consistent folds across models. We see that OLR is frequently the best performer, though the NB baseline is quite strong. By comparing with Table 1 we see that OLR usually gets higher QWK, relative to the training data, than two expert raters get between themselves. This is a common baseline in the AES field, but the comparison is somewhat flawed. The model QWKs measure agreement with the final scores which are, by design of the scoring process, less variable than an individual's scores. In contrast, the inter-rater QWKs compare two individuals. A better comparison would be the QWK achieved between two complete iterations of the scoring process, but we do not currently have that data.

We observe that the QWK of the length baseline is always below that of our OLR and NB models, but can reach nearly 0.8. This length correlation with score is unavoidable. However, especially as we want our scoring models to help drive feedback, it is important to ensure we are learning something about essay scoring *beyond* length. The fact that our models universally outperform the length-only baseline provides some assurance. Furthermore, in experiments outside the scope of this paper, we have found that predicted distributions from OLR can often differentiate between higher and lower scoring essays even when evaluating only a constrained 250-word beginning of each essay.

### 2.7 Practical considerations for deployment

High performing predictive models can be trained with several hundred scored essays. However, it is essential to acknowledge the sampling methodology used to collect the training data, and how that reflects the population using the AWE system. Frequently, initial training sets for prompts are easiest to collect from a single school or district. When we deploy such models more broadly, we are careful to monitor their performance for unexpected patterns. We often collect additional data from our user base and get it scored by experts, supplementing the original training data. We believe it to be important to treat scoring models as living algorithms, needing maintenance to stay at their best performance.

In practice, we consider a more diverse set of metrics than the QWK values presented here. The distribution and type of errors, (im)balance of score points in the training data, diversity of the training set, relative performance of the models for a prompt's several traits, and patterns of scoring of the new model on any previously submitted essays are all important.

## 3 FROM PREDICTIVE MODELS TO ESSAY FEEDBACK

Automated scoring brings value to the classroom in the form of whole-essay summative assessment. However, targeted formative feedback is vital to the development of writing proficiency [23, 28]. Research in writing education demonstrates that localized, actionable feedback, presented as part of an iterative writing process, is effective [14, 17]. By connecting comments to the rubric's evaluation criteria, students can use the feedback to foster their ability to reflect and self-assess [18, 20]. Self-reported responses to teacher feedback confirm that students value a combination of positive and critical comments that are specific to their own writing, and connected to the evaluation criteria [31]. Such feedback is especially valuable on preliminary drafts, instead of later in the writing process [13]. Even the most dedicated instructor would struggle to provide a set of high-quality targeted comments on every draft of every student's essay, for every prompt they assign throughout the year. However, it would be a mistake to consider only the labor-saving implications of AWE. Classroom observations of Revision Assistant suggest that automated essay feedback allows the teacher to step back from the sometimes-adversarial red pen and engage with their class as guides and readers, modeling the interpretation of feedback alongside their students.

Revision Assistant provides sentence-level formative feedback tied to the rubric traits for each draft a student submits. As described in the next section, the scoring models directly drive the selection of feedback targets. While the feedback text is authored by content experts, we require no specialized annotation at the sentence or phrase level, and no manual feature engineering to select comment-worthy sentences for each trait.

| | Genre | Grade | Len | Scored | Extra | Trait 1 OLR | 1 NB | 1 LEN | Trait 2 OLR | 2 NB | 2 LEN | Trait 3 OLR | 3 NB | 3 LEN | Trait 4 OLR | 4 NB | 4 LEN |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| P1 | Analysis | 11-12 | 382 | 457 | 1096 | **0.76** | 0.67 | 0.47 | **0.72** | 0.67 | 0.47 | **0.69** | 0.60 | 0.21 | **0.62** | 0.61 | 0.23 |
| P2 | Argument | 6-8 | 263 | 999 | 1183 | **0.82** | 0.78 | 0.57 | **0.80** | 0.76 | 0.58 | 0.78 | **0.80** | 0.51 | 0.74 | **0.75** | 0.54 |
| P3 | Argument | 9-12 | 387 | 399 | 548 | **0.76** | 0.72 | 0.65 | **0.76** | **0.76** | 0.64 | **0.76** | 0.75 | 0.64 | **0.71** | **0.71** | 0.49 |
| P4 | Argument | 11-12 | 382 | 755 | 940 | **0.77** | 0.75 | 0.65 | **0.77** | 0.73 | 0.64 | **0.75** | 0.72 | 0.62 | **0.72** | 0.69 | 0.55 |
| P5 | Informative | 7-8 | 294 | 634 | 3497 | **0.67** | 0.59 | 0.57 | **0.64** | 0.61 | 0.60 | **0.63** | 0.56 | 0.59 | **0.59** | 0.50 | 0.52 |
| P6 | Informative | 9-10 | 359 | 493 | 930 | **0.80** | 0.77 | 0.64 | **0.79** | 0.76 | 0.64 | **0.73** | 0.69 | 0.64 | **0.73** | 0.71 | 0.57 |
| P7 | Informative | 11-12 | 410 | 280 | 14 | 0.78 | **0.79** | 0.64 | **0.78** | 0.77 | 0.69 | **0.75** | 0.74 | 0.70 | **0.70** | 0.69 | 0.65 |
| P8 | Narrative | 6-6 | 301 | 283 | 377 | **0.84** | **0.84** | 0.77 | 0.82 | **0.84** | 0.77 | **0.81** | 0.84 | 0.77 | 0.78 | **0.79** | 0.75 |
| P9 | Narrative | 11-12 | 455 | 695 | 260 | **0.69** | 0.68 | 0.47 | 0.67 | **0.69** | 0.48 | **0.67** | 0.63 | 0.53 | **0.63** | 0.59 | 0.18 |

**Table 3: QWK performance on Revision Assistant datasets. Each row is a prompt. Prompts are scored on various 4-trait rubrics. OLR is our ordinal logistic regression model. NB and LEN are our Naive Bayes and length-only baseline models.**

### 3.1 Existing work

Several other AWE systems include automated scoring alongside some kind of specific feedback. In many cases, that feedback is not directly driven by the scoring algorithms themselves. For instance, ETS Criterion uses the scoring models from its e-rater system, but provides the student with feedback based on a series of separate algorithms that detect usage and mechanics errors, particular aspects of style (e.g. passive voice), and detection of discourse elements [5]. Writing Pal is an intelligent tutoring system that scaffolds writing and feedback within learning tasks [24]. Its automated holistic feedback targets student writing strategies. However, it too uses highly engineered essay features and a series of algorithms for each feedback type, divorced from the scoring model itself.

More similarly to our approach, Andersen et al. [3] describe a system to give model-driven sentence-level feedback focused on mechanics and usage. However, their models require sentence level annotations of grammatical errors. The authors assign pseudo scores to the sentences based on the whole essay score and the count of errors within the sentence. They train a model directly on these sentence-level scores, using it to predict the score of a new sentence. For our rubrics, obtaining sentence-level annotations for multiple rubric traits is prohibitively costly, even discounting the work of creating a validated and usable annotation guide.

Outside of the essay evaluation domain, there has been a recent surge of interest in generating rationales for the predictions of otherwise opaque systems such as deep neural networks. For instance, [16] describes a system for generating rationales for text categorization by finding small segments of target texts that still receive the same categorization. Our problem has a different framing, as we wish to identify both strong and weak components of a text, but is similar in spirit to this type of work.

### 3.2 Problem definition

Many components contribute to the student writing and feedback experience in Revision Assistant, most of which are beyond the scope of this paper. Here we address a particular part of the feedback process, namely selecting influential sentences to highlight.

When a student requests feedback, their draft is sent to our scoring models (one per rubric trait). The student receives the predicted scores in a visual and friendly format, mimicking wifi signals. In addition, we submit a series of modified versions of the essay to the models to estimate the influence that each modification has on the essay's score. Our modifications are at the sentence level, such that for each sentence we estimate a polarity (strong/weak) and magnitude of influence over each trait. We then select up to four of these sentences, optimizing for magnitude of influence against several constraints, and match these sentences to comments using the polarity, trait, essay score, and characteristics of the sentence and the essay. We present the feedback to the student by highlighting sentences as shown in Figure 1.

### 3.3 Essay modifications

Revising an essay is a process of incrementally altering the text to better match the demands of the prompt and rubric. To provide formative feedback to a student, we want to know which potential modifications to an essay will change the essay's score for the better, and guide the student to those revisions.

Our scoring models are certainly not causal, so we cannot directly look for modifications of the feature space (such as increasing the count of a particular n-gram) that would change the essay's score. Those feature edits would almost certainly correspond to unrealistic alterations in the English text, violating un-modeled linguistic structure. We also need to map the edit under consideration onto concrete suggestions to the student, who does not reason in n-grams. Instead, we first generate realistic edited versions of the essay and then ask how those edits alter the predicted score. Specifically, we look at versions of the essay where individual sentences have been omitted. If omitting a particular sentence has a particularly detrimental impact on the essay's score on a trait, that sentence is strong for that trait. Conversely, if removing a sentence actually improves the score, that sentence is weak. This simplification is a useful conceit for reasoning about texts through probability distributions, and is elaborated on in the next section.

The quality and types of feedback that these models can select is directly tied to the essay variants considered. Here we use sentence removal to generate variants, which is quite easy to implement. Recent research into revision extraction and classification [33] may offer a useful avenue for modeling and reasoning about actual revision practices in our growing corpus of student drafts.

Creating variants through removing sentences means that essay variants may be quite different in total length. We know that length

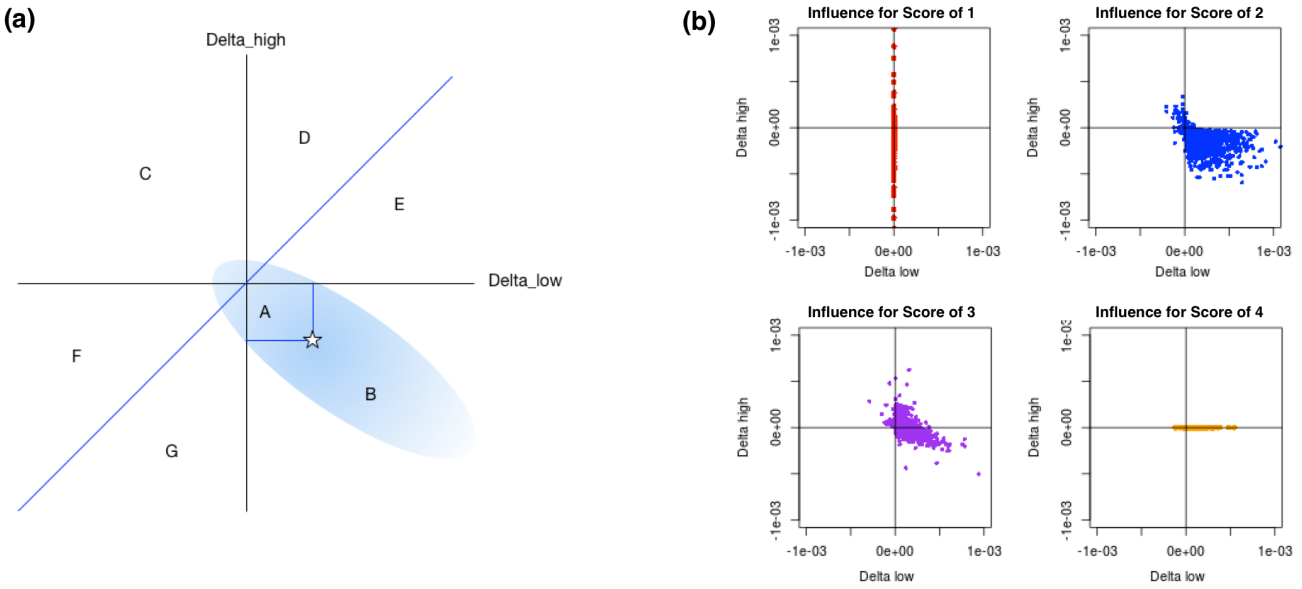


**Figure 2:**
**(a) Theoretical representation of influence between $\Delta_{high}$ and $\Delta_{low}$. The shaded region indicates a background distribution.**
**(b) Background distributions of essay modifications in influence space with real data across four score points on a single trait.**

is strongly correlated with essay scores, and this is invariably captured in supervised models. However, this can be an inaccurate inference at the sentence level. In an earlier iteration of our feedback system, using the NB algorithm, this property invariably led to very long sentences being identified as strong, and very short sentences as weak. In our current system, we include length as a feature in the OLR model and then artificially hold that feature constant between the essay variants. Since the parameters of the OLR model are jointly estimated, the remaining non-length features more accurately describe other components of strong or weak writing. We no longer observe the same overwhelming length effect on feedback.

## 3.4 Influence Space

Consider an essay $e$, represented by feature vector $\mathbf{x}$. Now consider one of a set of edits $d \in D$, which can be applied to an essay to change its text and therefore its feature representation. $d(\mathbf{x}) = \mathbf{x'}$. We also have a predictive model $M$ that, given a feature vector will return a predicted score distribution over the possible scores, $M(\mathbf{x}) = [Pr(1), Pr(2), Pr(3), Pr(4)]$. The *argmax* of this distribution is the predicted score $s*$. The difference between $M(\mathbf{x})$ and $M(\mathbf{x'})$ can be expressed by two numbers: the change in total probability mass on scores lower than $s*$, and the change in total probability mass on scores higher than $s*$. Call these numbers $\Delta_{low}$ and $\Delta_{high}$, respectively. When $\Delta_{low}$ is positive, for instance, the edit increased the probability of lower scores, decreasing the predicted quality of the essay.

We compute one $(\Delta_{low}, \Delta_{high})$ pair for each sentence, representing the sentence's contribution to the essay score distribution. This generates a set of two-dimensional points in what we will call *influence space*. Note that there are many ways that the shape of the distribution can change between $M(\mathbf{x})$ and $M(\mathbf{x'})$. This 2D projection captures the patterns we are most interested in while remaining simple enough to reason about.

Given this definition of influence space, we could declare that the distance from the origin represents the influence of a sentence, and the direction indicates the polarity of that influence. However, most sentences are better than nothing and have positive $\Delta_{low}$ and negative $\Delta_{high}$. That is, we *expect* that an arbitrary sentence will have a location in influence space somewhere in the lower right quadrant. Such a location does not indicate that the sentence is unusually influential.

To account for this expected influence of an arbitrary sentence, we use the influence space distributions of a large set of (potentially unscored) essays as a background distribution. When considering a new essay, we compare the influence space location of its sentences to this background distribution using Mahalanobis distance. We use this distance, rather than the distance from the origin, as the *influence* of the sentence. The direction of the point from the mean of the background distribution is its polarity.

Figure 2a shows an abstraction of influence space and a background distribution, with various regions labeled. Sentences in region B are more positively influential than the average sentence. These are sentences that are candidates for positive comments. Sentences in region C actually improve the essay by being absent. These are clear candidates for negative polarity comments. Sentences in region A contribute positively to the essay, but less than would be expected from an arbitrary sentence. We also treat these

as candidates for negative polarity comments. Sentences in regions D,E,F, and G either flatten or increase the peakiness of the predicted distribution. These could be candidates for specialized comments but are not currently deployed in Revision Assistant.

Figure 2b shows influence space plots for sentences in an authentic Revision Assistant data set. Each essay generates one point in the influence space per sentence it contains. Note that the distributions for 1 and 4 scores are degenerate, as it is not possible to receive lower or higher scores. In these cases, the background distribution flattens to a one-dimensional space in which A, B, and C are the only possible regions.

## 4 DEPLOYING AND EVALUATING FEEDBACK

The OLR based influence estimation system is deployed in Revision Assistant. We study a limited selection of live user data captured between October 2016 and May 2017. In this period, over 79,000 students used Revision Assistant. We evaluated an average of over 4,500 essay drafts per school day.

### 4.1 Considerations for deployment

We have described the problem of using whole-essay probabilistic models to estimate the influence of sentences on trait scores. Making this available to students in Revision Assistant involves solutions to a variety of problems beyond the scope of this paper. We note here several of the most relevant of these, as context for the use of the influence computations in the live product.

Complementing the model-driven sentence selection is a large body of comment texts written and curated by experienced writing instructors. Revision Assistant chooses from a structured pool of over a thousand such comments when presenting feedback to students. Matching influential sentences to these comments involves an additional suite of text processing tools (such as identifying quotations and dialogue) and heuristic labeling (such as dividing essays into structural components).

Revision Assistant starts with the estimated influence for each trait/sentence pair, and must choose a limited set of comments to show to a student. We therefore impose a variety of constraints, motivated by pedagogical and user experience considerations, on the number, specificity, and distribution of comments displayed on any particular essay draft. This defines a constrained optimization problem wherein we choose the optimal set of sentences such that the influence of the selected sentences is maximized while the constraints are respected as closely as possible.

Finally, the vast majority of students using Revision Assistant do so in good faith, and attempt to perform the writing task posed them. However, some students either misunderstand or intentionally veer from the task. Supervised scoring models should not be expected to detect or accurately score such off topic or bad faith essays, which will severely violate the distributional patterns of the training set. For this reason, we deploy several anomaly detectors to essays before scoring them or offering feedback. Anomalous essay detection is a challenging learning task in its own right, and beyond the scope of this paper, but it is an essential component of any practical AWE system.

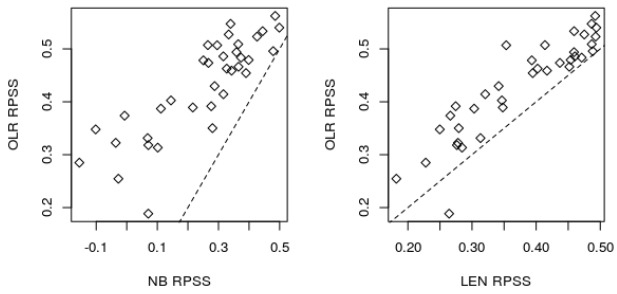


**Figure 3: Ranked Probability Skill Score for each prompt and trait, comparing the OLR to NB and LEN (larger values indicate larger increase over baseline). OLR is almost universally better calibrated than both NB and LEN.**

### 4.2 Evaluation

Standard baseline metrics for automated scoring, such as QWK, rely on scored examples treated as ground truth. Since a lack of sentence-level ground truth is central to our feedback problem formulation, the performance of feedback models cannot be measured with out-of-sample predictive accuracy. This necessitates a more diverse set of metrics. Here we present a suite of evaluations ranging from assumption checking and distributional characteristics of predictions, to offline blind evaluation of sentence selection, to observational metrics on live student usage of Revision Assistant.

*4.2.1 Model characteristics.* Feedback generation is a problem closer to inference than prediction, and as such, the assumptions and fit of the models have increased relevance. As presented in Section 2, our predictive model acknowledges the ordinal categorical nature of the essay scores. We generate feedback using realistic essay variants, as we cannot assume that the true causal correlation structure of language is reflected in our models. Finally, since we use small differences between predicted distributions to estimate influence, we choose an AES model that produces calibrated predicted score distributions.

In pure score prediction, the maximum of the predicted distribution is important but the shape of the distribution is otherwise irrelevant. For instance, the Naive Bayes model performs well but tends to predict degenerate distributions with a single score receiving a predicted probability of nearly 1. However, essay modifications rarely shift predictions by a whole score point, and our influence computations consider the shape of the entire predicted distribution. We want it to be meaningful, not just have its maximum at the right place.

The calibration of probabilistic predictions can be measured in a variety of ways [15]. One of these, appropriate for discrete ordinal predictions, is the Ranked Probability Score (RPS) [12]. For $N$ essays and $S$ possible score points,

$$RPS = \frac{1}{N}\sum_{n=1}^{N}\sum_{s=1}^{S}(P_{ns} - O_{ns})^2 \tag{4}$$

where $P_{ns} = \sum_{j=1}^{n} p_{js}$ is the predicted CDF and $O_{ns}$ is the empirically observed CDF (always a step function) for essay $n$. By using

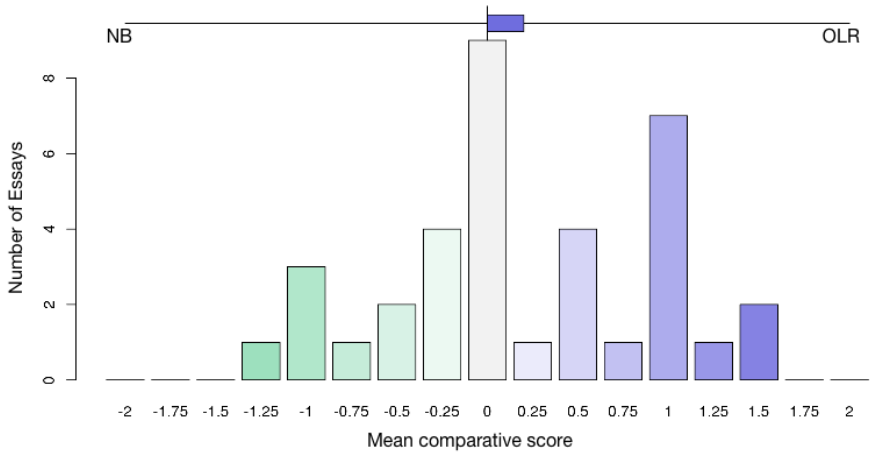


**Figure 4: Whole-essay judgements of sentence selection quality, with participants blind to the specific feedback algorithm. The x axis scale runs from strongly preferring NB generated comments (-2) to strongly preferring OLR generated comments (2). Scores from all annotators are averaged per essay.**

the CDFs, this measure penalizes a prediction for placing probability mass farther from the observed score. RPS is a proper scoring rule, meaning that it is optimized by the predictions that match the true generating distribution of the events under consideration.

Since the RPS depends on the distribution of the dataset as well as the predictions, direct comparisons across datasets are not appropriate. Instead, we use the Ranked Probability Skill Score, the ratio of the RPS of each algorithm to the RPS of a default algorithm. We use as default the algorithm that always predicts the baseline prior probabilities of each score. Figure 3 shows the RPSS values for our OLR model compared to the NB and LEN baselines for each trait in our example datasets. OLR is almost universally better calibrated than both baselines.

We note that the calibration metric is not explicitly related to feedback, and is in fact simply a measure of properties of the predictive models. Nevertheless, characteristics such as calibration are important prerequisites for believable feedback performance.

*4.2.2 Offline evaluations.* The deployed feedback system in Revision Assistant has many moving parts. To evaluate the influence estimation specifically, it is helpful to remove as many of those extraneous layers as possible. However, it is also difficult to evaluate the influence values directly. There are no established rubrics for hand scoring essays at the sentence level. Instead, we choose to evaluate the set of 4 sentences chosen by our influence estimation and feedback coordination systems. We do so with a blind comparative judgement task. The annotator sees the same essay on a split screen. Sentences are highlighted in each copy of the essay using one of the algorithms being compared. The trait and direction of influence for each highlighted sentence ("this is a strong sentence for the Language trait") are also displayed. The annotator then chooses whether they strongly prefer or moderately prefer one side, or consider the two comment sets equivalent. The two algorithms being compared are shown on random sides for each example, and the annotator is blind to which algorithm generated which set.

We performed a small such annotation task with 50 example essays and three Revision Assistant team members as annotators, comparing NB based sentence selections to OLR based sentence selections. The essay level results are shown in figure 4. We found that the annotators tended to prefer the OLR based selections. We also had the annotators indicate the quality of each individual comment (on a 5 point Likert scale). Though the results are not included here, we found that the OLR algorithm was slightly preferred on the comment level, and the NB comments were more frequently judged as extremely poor. Finally, we compared the essay level judgements to the sentence level judgements. Though they were strongly correlated, annotators did sometimes prefer a comment set despite preferring the individual comments in the other. We hypothesize that this indicates interactions between the comments - for instance multiple highlights working well together or a lack of diversity in highlights being judged as undesirable.

This offline comparative judgement task is helpful to us when deciding which algorithms to deploy. However, performing such an experiment at scale with trained, independent annotators would be prohibitively expensive.

*4.2.3 Live behavioral metrics.* We deployed the OLR based prediction and influence estimation described here to students in October 2016. Between 10/16 and 5/17, Revision Assistant gave scores and feedback on over 937,000 valid drafts from more than 79,000 students. Though controlled efficacy studies of Revision Assistant are still in progress, we can learn a lot from observational study of these users' behavior.

Students using Revision Assistant wrote an average of 7.7 drafts per essay. 89% of students wrote at least two drafts, while 58% of students wrote at least 4. After each draft, the students received

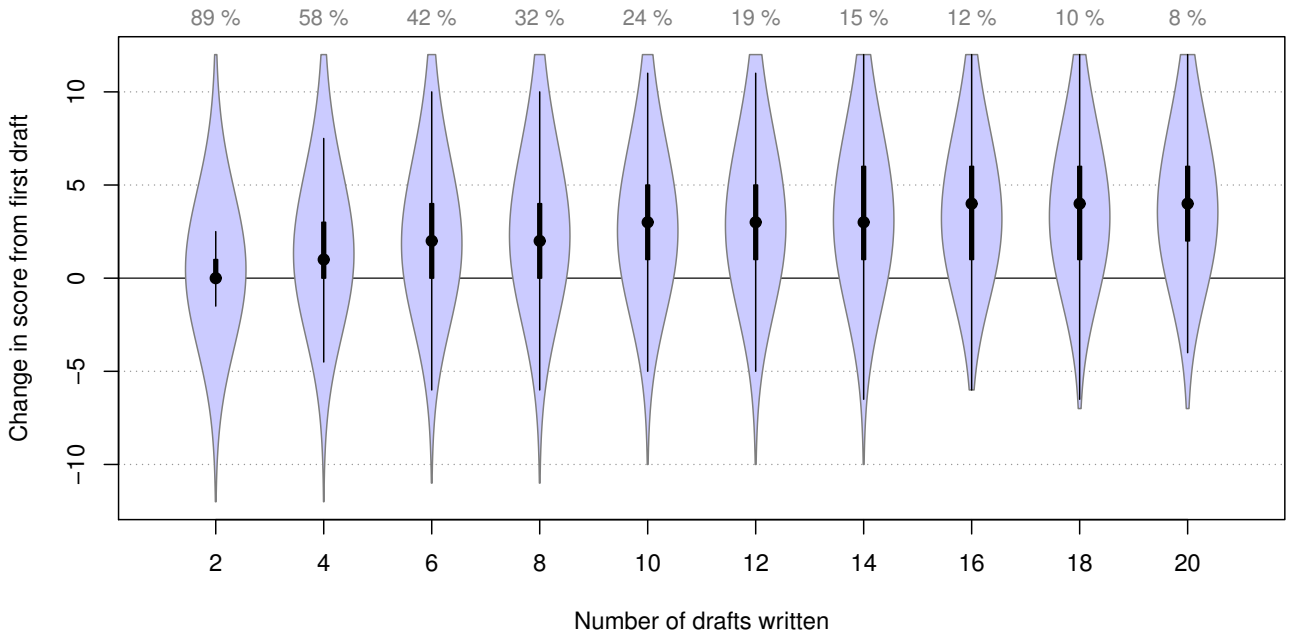


**Figure 5: Change in summed predicted score after N valid drafts, as compared to the first valid draft. Violin plots show the mean and center quartiles as the thicker black bars and kernel density estimates as the blue regions. Percentages above each plot indicate the fraction of essays with at least that many valid drafts.**

predicted scores as well as feedback comments. In general, students improved their score over the course of their revision process. Summed across all essay traits, students improved their score only slightly on their second draft (an average of 0.6 points). However, after 7 drafts (nearly the mean number), students improved by an average of 2.6 points. Twenty percent of students wrote at least 11 drafts, improving by an average of 3.2 points by that point. We note that students would probably improve their writing by revising many times without feedback, but believe that the Revision Assistant feedback is effective at encouraging revision and guiding the student toward influential changes. We also note that these score improvements are measured by our predictive algorithm, not independent scorers. Even so, this indicates that the model-driven feedback is eliciting edits that move essays toward higher predicted scores – the purpose of influence estimation as defined here.

Revision Assistant gives students the option of rating comments they receive as helpful or not helpful (see figure 1). Only about 8% of the 3.5 million comments that we displayed to students were marked in this way. The fraction of those marked as helpful is uneven between positive valence comments ("this is a good example of X") and negative valence comments ('you need to improve this'). Students overwhelmingly mark positive valence comments as helpful (88% helpful). We believe this to be largely driven by students appreciating praise (as in [13]), and in general don't see very many interesting patterns in the marks on positive valence comments. In contrast, negative valence comments are marked as helpful 72% of the time. Though negative valence comments are marked as helpful at a lower rate, we believe these marks are more indicative of the helpfulness of the comment.

We observe that the fraction of comments marked as helpful varies significantly across traits and prompts. We can use this information to draw attention to feedback models, comment texts, or other system features that might need attention. More relevant to the evaluation of our influence estimation process, after accounting for trait effects, we observe a significant positive correlation of our estimated influence value for a comment on whether that comment was rated as helpful by the student (given it was rated at all). Figure 6 shows this relationship.

## 5 CONCLUSIONS AND NEXT STEPS

Automated Writing Evaluation tools such as Revision Assistant can multiply the impact of dedicated teachers by providing more frequent formative feedback to students and softening the potentially adversarial interpretation of constructive criticism. Our ordinal logistic regression essay scoring model achieves state of the art predictive performance while preserving characteristics suitable for a novel sentence influence estimation task. We have introduced a model-driven influence estimation process that identifies sentences within an essay that deserve feedback without any sub-essay-level annotations. The automated feedback task does not lend itself to

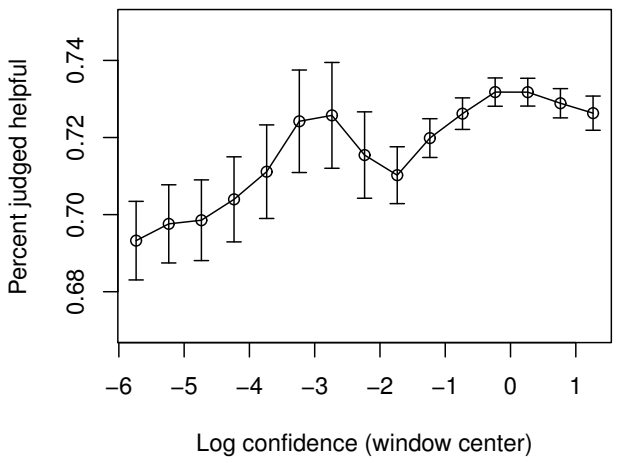


**Figure 6: For negative polarity comments, our measure of influence is significantly correlated with the percent of comments rated as 'helpful'. Points show the percentage of comments rated as helpful in a sliding window of log confidence values of width 1.5. Error bars show the 95% binomial proportion confidence interval. The rise and dip in the middle of the range are likely related to a discrete shift in our comment coordination heuristics for different regions of the influence space.**

single benchmark evaluations, so we have presented a diverse array of evaluation techniques for our approach. Our behavioral metrics indicate that Revision Assistant, driven by our influence estimation method, generates automated feedback that is both well-received by student writers and aligned with improving scores. Controlled efficacy studies currently in progress will supplement these metrics.

Future work in the space of automated writing evaluation may be informed by these results. We hope to see increased use of learned models for solving problems in writing evaluation that extend beyond score prediction. In the specific domain of influence estimation, we expect that research in revision extraction will lead to additional realistic methods of essay modification. Finally, we believe that additional gains could be made by combining the models for a rubric's traits. Trait scores tend to be highly correlated, and a joint model might be able to better describe the shared and unique components of writing, choosing feedback targets more specific to each trait.

## REFERENCES


[1] Alan Agresti. 2010. *Analysis of Ordinal Categorical Data* (2 ed.). John Wiley & Sons.

[2] D Alikaniotis, H Yannakoudakis, and M Rei. 2016. Automatic Text Scoring Using Neural Networks. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics* (2016), 715–725. arXiv:1606.04289

[3] Øistein E Andersen, Helen Yannakoudakis, Fiona Barker, and Tim Parish. 2013. Developing and testing a self-assessment and tutoring system. In *Innovative Use of NLP for Building Educational Applications*.

[4] Kellie J Archer. *glmnetcr : An R Package for Ordinal Response Prediction in High-dimensional Data Settings*. Technical Report. http://www.people.vcu.edu/

[5] Yigal Attali. 2004. Exploring the feedback and revision features of Criterion. In *Meeting of the National Council on Measurement in Education (NCME)*.

[6] Jing Chen, James H. Fife, Isaac I. Bejar, and André A. Rupp. 2016. Building e-rater ® Scoring Models Using Machine Learning Methods. *ETS Research Report Series* June (2016), n/a–n/a. https://doi.org/10.1002/ets2.12094

[7] M Chodorow and J Burstein. 2004. *Beyond Essay Length: Evaluating e-rater®'s Performance on TOEFL® Essays*. Technical Report February.

[8] S R Cole and C V Ananth. 2001. Regression models for unconstrained, partially or fully constrained continuation odds ratios. *International journal of epidemiology* 30, 6 (2001), 1379–82. https://doi.org/10.1093/ije/30.6.1379

[9] Ronan Cummins, Meng Zhang, and Ted Briscoe. 2016. Constrained Multi-Task Learning for Automated Essay Scoring. In *ACL 2016*. 789–799.

[10] Semire Dikli. 2006. An Overview of Automated Scoring of Essays. *Journal Of Technology Learning And Assessment* 5, 1 (2006), 2006–12. http://www.jtla.org

[11] Fei Dong and Yue Zhang. 2016. Automatic Features for Essay Scoring âĂŞ An Empirical Study. *EMNLP* 1966 (2016), 1072–1077.

[12] Edward S. Epstein. 1969. A Scoring System for Probability Forecasts of Ranked Categories. (1969), 985–987 pages. https://doi.org/10.1175/1520-0450(1969)008<0985:ASSFPF>2.0.CO;2

[13] Dana R. Ferris. 1995. Student Reactions to Teacher Response in Multiple-Draft Composition Classrooms. *TESOL Quarterly* 29, 1 (1995), 33–53.

[14] Dana R. Ferris. 1997. The Influence of Teacher Commentary on Student Revision. *TESOL Quarterly* 31, 2 (1997), 315–339. https://doi.org/10.2307/3588049

[15] Tilmann Gneiting and Matthias Katzfuss. 2014. Probabilistic Forecasting. *Annual Review of Statistics and Its Application* 1 (2014), 125–151.

[16] Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2016. Rationalizing Neural Predictions. In *EMNLP 2016*. 107–117. arXiv:1606.04155 http://arxiv.org/abs/1606.04155

[17] Robin Martin. 2011. Rhetoric of teacher comments on student writing. *Young Scholars in Writing* 8 (2011), 16–29.

[18] Hedy McGarrell and Jeff Verbeem. 2007. Motivating revision of drafts through formative feedback. *ELT Journal* 61, 3 (2007), 228–236.

[19] Danielle S. McNamara, Scott A. Crossley, Rod D. Roscoe, Laura K. Allen, and Jianmin Dai. 2015. A hierarchical classification approach to automated essay scoring. *Assessing Writing* 23 (2015), 35–59.

[20] David J Nicol and Debra Macfarlane-Dick. 2006. Formative assessment and self-regulated learning : A model and seven principles of good feedback practice. *Studies in Higher Education (2006)*, 31, 2 (2006), 199–218.

[21] Ellis B Page. 1966. The imminence of... grading essays by computer. *The Phi Delta Kappan* (1966), 238–243.

[22] Peter Phandi. 2015. Flexible Domain Adaptation for Automated Essay Scoring Using Correlated Linear Regression. *EMNLP* September (2015), 431–439.

[23] Rod D. Roscoe, Erica L. Snow, and Danielle S. McNamara. 2013. Feedback and revising in an intelligent tutoring system for writing strategies. *Lecture Notes in Computer Science* 7926 LNAI (2013), 259–268.

[24] Rod D Roscoe, Laura K Varner, Scott A Crossley, and Danielle S Mcnamara. 2013. Developing pedagogically-guided algorithms for intelligent writing feedback. *Int. J. Learning Technology* 8, 4 (2013), 362–381. ftp://129.219.222.66/pdf/RoscoeEtAl

[25] Lm Rudner and Tahung Liang. 2002. Automated essay scoring using Bayes' theorem. *The Journal of Technology, Learning and ...* 1, 2 (2002).

[26] Mark D. Shermis. 2014. State-of-the-art automated essay scoring: Competition, results, and future directions from a United States demonstration. *Assessing Writing* 20 (2014), 53–76. https://doi.org/10.1016/j.asw.2013.04.001

[27] Mark D. Shermis and Jill C Burstein (Eds.). 2002. *Automated Essay Scoring: A Cross-disciplinary Perspective* (1 ed.). Routledge.

[28] Valerie J Shute. 2008. Focus on Formative Feedback. *Review of Educational Research* 78228173, 1 (2008), 153–189.

[29] Kaveh Taghipour and Hwee Tou Ng. 2016. A Neural Approach to Automated Essay Scoring. In *EMNLP 2016*. 1882–1891.

[30] Kristina Toutanova, Dan Klein, and Christopher D Manning. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. *NAACL* (2003).

[31] Melanie R. Weaver. 2006. Do students value feedback? Student perceptions of tutors' written responses. *Assessment & Evaluation in Higher Education* 31, 3 (2006), 379–394. https://doi.org/10.1080/02602930500353061

[32] Torsten Zesch, Michael Wojatzki, and Dirk Scholten-Akoun. 2015. Task-Independent Features for Automated Essay Grading. *Building Educational Applications Workshop at NAACL* June (2015), 224–232.

[33] Fan Zhang and Diane Litman. 2015. Annotation and Classification of Argumentative Writing Revisions. (2015), 133–143.