Integrating educational measurement practices with automated scoring using *RSMTool*

Anastassia Loukina and **Nitin Madnani**
Educational Testing Service
660 Rosedale Rd
Princeton, NJ, 08541 USA
{aloukina,nmadnani}@ets.org

1 Introduction

We present a suite of tools for building and evaluating statistical models for automatically scoring spoken and written responses.

Automated scoring of constructed spoken and written responses is a multi-stage process. First, the responses are processed to extract a set of features representing different dimensions relevant to a given assessment. These features are then combined in a statistical model which maps feature values to the final score (Page, 1966; Neumeyer et al., 1996; Burstein et al., 1998; Landauer et al., 2003; Zechner et al., 2009; Bernstein et al., 2010).

The development of automated scoring models, especially ones used for operational scoring of test taker responses, is, in most cases, the result of collaboration between NLP (or Speech) researchers and experts in educational measurement. While the former work on feature development and machine learning models, the latter ensure that the final model is in line with the established desiderata for assessment design, validity and fairness (Yang et al., 2002; Clauser et al., 2002; Williamson et al., 2012).

We present RSMTool (Rater Scoring Model Tool), an open-source tool[1] that we have developed to aid NLP researchers working on new features or components for automated scoring models. RSMTool helps researchers evaluate to what extent their proposed changes to the scoring model meet the guidelines developed by the educational measurement community. RSMTool not only automates the model building and evaluation process to ensure to consistency between different researchers working on the same model, but also produces a detailed report which integrates measurement guidelines and quantifies how well the model meets them.

[1] https://github.com/EducationalTestingService/rsmtool

The measurement guidelines currently implemented in RSMTool follow the framework suggested by Williamson et al. (2012). It was developed for the evaluation of e-rater, an automated essay scoring engine (Attali and Burstein, 2006), but is generalizable to other applications of automated scoring. This framework was chosen because it offers a comprehensive set of criteria that cover not only the accuracy of predicted scores but also other aspects such as fairness of the automated scoring engine.

One of the main strength of RSMTool is its flexibility. Not all the recommendations made by Williamson et al. (2012) are universally accepted by the automated scoring community. For example, Yannakoudakis and Cummins (2015) recently proposed a different approach to evaluating accuracy of automated scoring. The structure of RSMTool makes it easy for researchers to add new analyses without making any changes to the core code structure thus allowing for a wide range of psychometric evaluations.

Finally, RSMTool treats scoring as a regression of the final score on a set of non-sparse numeric features. To this extent, it provides access to multiple regression algorithms including linear and non-linear models. However, this is by far not the only approach to automated scoring. See, for example, Chen and He (2013) and Shermis (2014a) for an overview. To address this, RSMTool incorporates a separate tool named RSMEval for evaluating predictions generated by external systems. This tool is described in more detail in §5.

In the rest of the document, we introduce the data processing pipeline implemented in RSMTool and then focus on the report presented to the user. As an example, we use the data from the The Hewlett Foundation competition on Automated Essay Scoring (Shermis, 2014a) and build a scoring model using simple features inspired by the ones

described by Attali and Burstein (2006).[2] Note that the scoring system we build is entirely for illustration purposes. The complete report automatically generated by RSMTool is available at: `http://bit.ly/rsmtool`.

2 Data Preprocessing

RSMTool takes as input the data files that contain feature values and observed scores for the training and evaluation partitions. Since the tool does not include any text or audio processing components, all feature computation (as well as data partitioning) must be done beforehand.

The supplied features are then pre-processed to ensure the reliability of the final model. First, all responses with non-numeric human scores or feature values are removed from the data. For the remaining responses, outliers in each column are appropriately truncated to the mean feature value $\pm$ 4 standard deviations (cf. Zechner et al. (2009)). Finally, all feature values are standardized into z-scores.

To ensure generalization, any parameters used for feature standardization and transformation are computed only on the training partition and then applied to the evaluation partition.

3 Model building

The processed feature values are used to train a statistical model that maps feature values to human scores. The features used to train the chosen model can either be pre-defined or automatically selected.

RSMTool allows the use of simple OLS regression as well as several more sophisticated regressors including Ridge, SVR, AdaBoost, and Random Forests, available through the SKLL toolkit.[3] The tool also includes several regressors which ensure that all coefficients in the final model are positive to meet the requirement that all feature contributions are additive (Lipovetsky, 2009). These are non-negative least squares regression (NNLS) (Lawson and Hanson, 1981) and a constrained version of Lasso regression (Goeman, 2010).

The chosen model is trained on the training partition and the parameters learned are then used to generate predictions on the pre-processed feature values of the responses from the evaluation partition.

4 Model evaluation report

As part of its output, RSMTool automatically generates a comprehensive HTML report describing all aspects of the model training process.

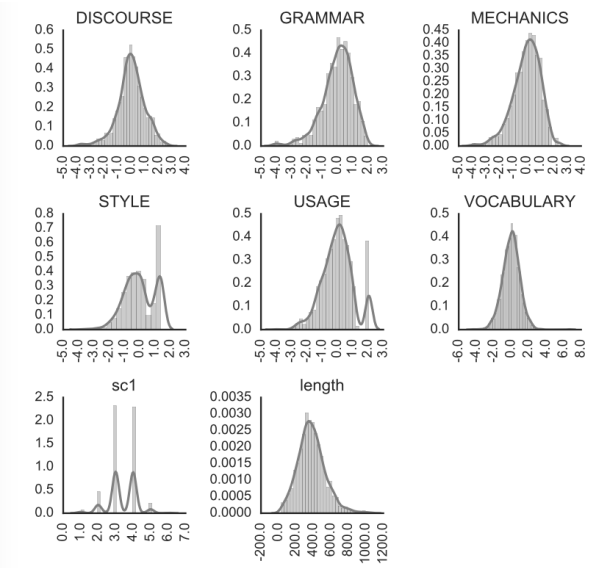

Figure 1: A graph from the ASAP RSMTool report showing distributions of the pre-processed feature values, the human score (`sc1`), and response length. The graph shows histograms as well as a univariate kernel density estimate, obtained using a gaussian kernel (denoted by the smoothed line).

Each section contains tables and plots with information crucial to conducting a well-rounded evaluation of the scoring model. All numeric results are saved as CSV files and can be used directly for more complex analyses. All sections of the report integrate the recommendations following the Williamson et al. (2012) framework; values that do not meet these recommendations are either automatically highlighted in red (in tables) or indicated graphically (in plots).

Although the report contains several sections, there are three salient ones that we will focus on in this document: data description, model summary, and model performance.

4.1 Data Description

The data description part of the report allows the user to review various distributional properties of the features used to train the model. These include descriptive statistics such as mean, range, the over-

[2] The data from the competition is publicly available at `https:/www.kaggle.com/c/asap-aes/data/`.

[3] `github.com/EducationalTestingService/skll`

	sc1	length	DISCOURSE	GRAMMAR	MECHANICS	STYLE	USAGE	VOCABULARY
sc1	1.000	0.649	0.609	0.316	0.504	0.525	0.266	0.312
length	0.649	1.000	0.814	0.213	0.387	0.611	0.114	0.066
DISCOURSE	0.609	0.814	1.000	0.196	0.372	0.557	0.105	0.047
GRAMMAR	0.316	0.213	0.196	1.000	0.399	0.217	0.327	0.171
MECHANICS	0.504	0.387	0.372	0.399	1.000	0.309	0.365	0.353
STYLE	0.525	0.611	0.557	0.217	0.309	1.000	0.124	0.179
USAGE	0.266	0.114	0.105	0.327	0.365	0.124	1.000	0.232
VOCABULARY	0.312	0.066	0.047	0.171	0.353	0.179	0.232	1.000

Figure 2: A table from the ASAP RSMTool report showing inter-feature correlations as well the correlation of each feature with the human score and response length. RSMTool automatically highlights in red the values that Williamson et al. (2012) consider too high or too low.

all distributions of the feature values, and their correlations with human scores and all other features. For example, Figures 1 and 2 are excerpts from our sample ASAP RSMTool report showing the feature distributions and the inter-feature correlations, respectively.

The effect of response length on automated scoring is a controversial topic (Perelman, 2014; Shermis, 2014b). Therefore, when evaluating a new feature one must always consider its relationship (latent or otherwise) to response length. RSMTool allows the user to optionally specify a column containing response length in the original data and, if so specified, automatically computes the marginal correlations between each feature and length as well as partial correlations between each feature and the human score after controlling for length. This helps clearly bring out the contribution that a new feature makes to the scoring model above and beyond being a proxy for response length.

Another important aspect of a scoring model used in high-stakes assessments is whether it behaves differently for different subgroups present in the data (Bridgeman et al., 2012). These subgroups can be defined in several ways e.g., the demographics of the test-taker population, or the different questions in a test. RSMTool helps examine this aspect by plotting the feature distributions for each subgroup column identified by the user in the data.

4.2 Model Summary

Interpretation of the final statistical model itself is also an important consideration in assessment context. In order to make an argument about validity of the automated scoring, we should be able to explain how each measured construct contributes to the final test score (e.g., see Bernstein et al. (2010) for discussion).

The "Model summary" section presents all the information necessary to evaluate the model by presenting the learned parameters in an easy to understand manner. For linear models, this section includes the standardized coefficients as well as relative coefficients, presented both in a table as well as graphically, as shown in Figure 3.

feature	standardized	relative
DISCOURSE	0.407	0.375
GRAMMAR	0.071	0.065
MECHANICS	0.181	0.166
STYLE	0.187	0.173
USAGE	0.073	0.067
VOCABULARY	0.166	0.153

Here are the same values, shown graphically.

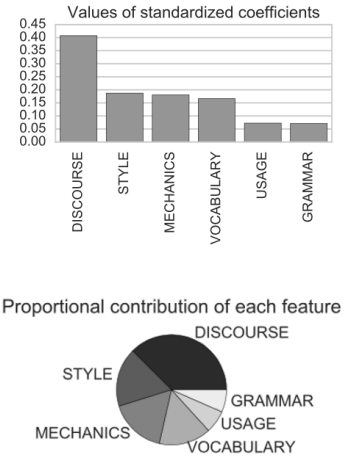

Figure 3: An excerpt from the model summary section of the ASAP RSMTool report.

4.3 Model performance

This section of the report focuses on the accuracy of the trained model's predictions on the responses in the evaluation partition. It does so by comparing them to the scores assigned to those same responses by expert human raters.

When producing statistics for the model performance, it might be useful to consider not just the raw scores produced by the chosen regression model but also (a) scores that are rescaled to match the human score distribution on the training set, (b) scores that are trimmed to be in the score range acceptable for the item, and (c) scores rounded to the nearest integers. All of these score transformations are commonly used in the automated scoring literature and, in some instances, can even be applied in combination.

RSMTool computes six different versions of scores and, for each such type, computes all standard evaluation metrics recommended by (Williamson et al., 2012) for automated scoring models such as Pearson's correlation, quadratic weighted kappa, and percentage agreement. RSMTool also computes measures designed specifically for evaluating automated scoring such as the standardized mean difference between human and automated scores which ensures that both sets of scores are centered on the the same point (Williamson et al., 2012).

This section also presents additional information for evaluating model performance in the form of confusion matrices and a comparison of the distributions of the human and predicted scores. As in the other sections of the report, the values which fall outside the recommended ranges are highlighted in red or indicated graphically.

Another important aspect of model performance is to compare the agreement of its predictions with an expert against how well two experts' scores agree with each other. If two humans cannot agree on how to score a given set of essays, it is not reasonable to expect a statistical model to learn how to do the same. RSMTool allows the user to optionally specify a column in the data containing the scores from a second expert, and, if so specified, automatically computes the agreement between two human raters and the corresponding degradation for machine scores. Figure 4 shows how RSMTool compares human-machine agreement statistics to human-human statistics.

Finally, this section computes all of the evaluation metrics for each subgroup defined in the data, as already motivated in the Data Description section. Figure 5 shows an example of this from our sample RSMTool report.[4]

4.4 Customization

All the settings for an experiment are supplied to RSMTool as a self-contained configuration file in JSON format. This ensures the reproducibility of the experiments and allows the user to customize the analyses depending on data availability.

Furthermore, RSMTool has been designed to make it easy for the user to customize the final HTML report. Each section of the report is implemented as a separate IPython notebook (Pérez and Granger, 2007). The user can decide which sections should be included into the final HTML report and in which order.

This approach also makes it very easy to add new evaluations. For example, a researcher who wants to use different evaluation metrics instead of, or in combination with, the existing evaluations, would only need to create an additional IPython notebooks containing custom analyses and have them included into the final report along with the other sections.

5 RSMEval and RSMPredict

In addition to RSMTool that covers the full preprocess-train-predict-evaluate pipeline, we also provide two additional tools that cover only the evaluation and the prediction parts of the pipeline.

The first, RSMPredict, can generate predictions for new data based on an existing scoring model. This is particularly useful in scenarios where a researcher might want to generate predictions for newly obtained test-taker data using a model that is already in operation.

The second tool, RSMEval, can take as input already existing predictions and generate an HTML report similar to the one described for the main tool. This tool is particularly useful in scenarios where the researcher would prefer to use a more sophisticated or complex machine learning algorithm to train the model and produce the predictions but would still like to see a comprehensive evaluation of the predictions, including subgroups.

Both RSMPredict and RSMEval are compatible with the outputs of RSMTool and use similar interfaces and configuration files.

[4]Note that the original ASAP data does not contain any demographic information, such as the test-taker's native language (L1), we randomly assigned an L1 value to each test-taker for illustration purposes.

N	h1_mean	h1_sd	h1_min	h1_max	h2_mean	h2_sd	h2_min	h2_max	corr	wtkappa	kappa	exact_agr	adj_agr
594.000	3.434	0.775	1.000	6.000	3.424	0.776	1.000	6.000	0.795	0.795	0.623	76.431	99.663

(a) Agreement statistics between two expert raters.

		corr	kappa	wtkappa	exact_agr	adj_agr	SMD
raw	H-H	0.795	0.623	0.795	76.431	99.663	-0.013
	H-M	0.722	0.421	0.627	0.000	94.949	-0.062
	diff	-0.073	-0.202	-0.168	-76.431	-4.714	-0.049
raw_trim	H-H	0.795	0.623	0.795	76.431	99.663	-0.013
	H-M	0.722	0.421	0.627	0.000	94.949	-0.062
	diff	-0.073	-0.202	-0.168	-76.431	-4.714	-0.049
raw_trim_round	H-H	0.795	0.623	0.795	76.431	99.663	-0.013
	H-M	0.639	0.421	0.627	65.320	98.990	-0.047
	diff	-0.156	-0.202	-0.168	-11.111	-0.673	-0.034
scale	H-H	0.795	0.623	0.795	76.431	99.663	-0.013
	H-M	0.722	0.403	0.672	0.000	91.246	-0.061
	diff	-0.073	-0.221	-0.123	-76.431	-8.418	-0.048
scale_trim	H-H	0.795	0.623	0.795	76.431	99.663	-0.013
	H-M	0.722	0.405	0.671	0.000	91.246	-0.060
	diff	-0.073	-0.218	-0.124	-76.431	-8.418	-0.047
scale_trim_round	H-H	0.795	0.623	0.795	76.431	99.663	-0.013
	H-M	0.672	0.405	0.671	62.795	99.158	-0.043
	diff	-0.124	-0.218	-0.124	-13.636	-0.505	-0.030

(b) Degradation statistics when comparing human-machine agreement to agreement between two expert raters. RSMTool provides these statistics not just for the `raw` scores but also for the scaled, trimmed, and rounded versions. It also automatically highlights degradations that might be considered too high as per Williamson et al. (2012).

Figure 4: Two tables from the consistency section of the ASAP RSMTool report.

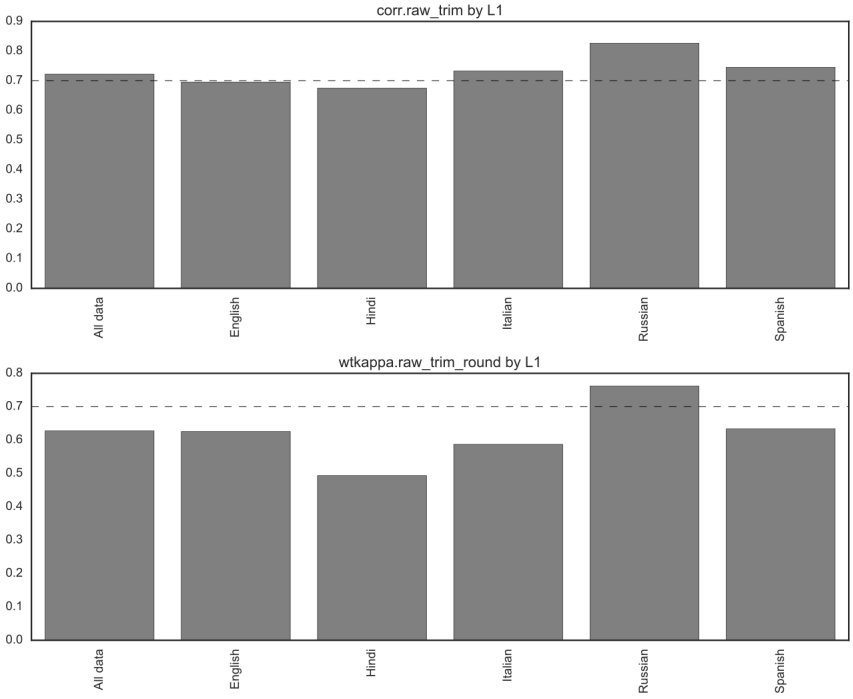

Figure 5: Graphs from the ASAP RSMTool report showing Pearson's correlations and quadratic weighted kappa by the artificially generated L1 subgroup, denoting the test-takers' native language. RSMTool graphically indicates the values that are above 0.7, as per Williamson et al (2012).

References

Yigal Attali and Jill Burstein. 2006. Automated essay scoring with e-rater V. 2. *The Journal of Technology, Learning and Assessment*, 4(3):1–30.

Jared Bernstein, A. Van Moere, and Jian Cheng. 2010. Validating automated speaking tests. *Language Testing*, 27(3):355–377.

Brent Bridgeman, Catherine Trapani, and Yigal Attali. 2012. Comparison of Human and Machine Scoring of Essays: Differences by Gender, Ethnicity, and Country. *Applied Measurement in Education*, 25(1):27–40.

Jill Burstein, Karen Kukich, Susanne Wolff, Chi Lu, Martin Chodorow, Lisa Braden-Harder, and Mary Dee Harris. 1998. Automated scoring using a hybrid feature identification technique. In *Proceedings of the 36th annual meeting on Association for Computational Linguistics - and 17th International Conference on Computational Linguistics*, volume 1, pages 206–210. Association for Computational Linguistics.

Hongbo Chen and Ben He. 2013. Automated Essay Scoring by Maximizing Human-machine Agreement. *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1741–1752.

Brian E. Clauser, Michael T. Kane, and David B. Swanson. 2002. Validity Issues for Performance-Based Tests Scored With Computer-Automated Scoring Systems. *Applied Measurement in Education*, 15(4):413–432.

Jelle J. Goeman. 2010. L1 penalized estimation in the Cox proportional hazards model. *Biometrical journal*, 52(1):70–84.

Thomas K. Landauer, Darrell Laham, and Peter W. Foltz. 2003. Automatic Essay Assessment. *Assessment in Education: Principles, Policy & Practice*, 10(3):295–308.

Charles L. Lawson and Richard J. Hanson. 1981. *Solving Least Squares Problems*. Prentice-Hall, Englewood Cliffs, NJ.

Stan Lipovetsky. 2009. Linear regression with special coefficient features attained via parameterization in exponential, logistic, and multinomiallogit forms. *Mathematical and Computer Modelling*, 49(7-8):1427–1435.

Leonardo Neumeyer, Horacio Franco, Mitchell Weintraub, and Patti Price. 1996. Automatic text-independent pronunciation scoring of foreign language student speech. *Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP '96*, 3:1457–1460.

Ellis B. Page. 1966. The imminence of... grading essays by computer. *The Phi Delta Kappan*, 47(5):238–243.

Les Perelman. 2014. When the state of the art is counting words. *Assessing Writing*, 21:104–111, jul.

Fernando Pérez and Brian E. Granger. 2007. IPython: a system for interactive scientific computing. *Computing in Science and Engineering*, 9(3):21–29.

Mark D. Shermis. 2014a. State-of-the-art automated essay scoring: Competition, results, and future directions from a United States demonstration. *Assessing Writing*, 20:53–76.

Mark D. Shermis. 2014b. The challenges of emulating human behavior in writing assessment. *Assessing Writing*, 22:91–99.

David M. Williamson, Xiaoming Xi, and F. Jay Breyer. 2012. A framework for evaluation and use of automated scoring. *Educational measurement: issues and practice*, 31(1):2–13.

Yongwei Yang, Chad W. Buckendahl, Piotr J. Juszkiewicz, and Dennison S. Bhola. 2002. A Review of Strategies for Validating Computer-Automated Scoring. *Applied Measurement in Education*, 15(4):391–412, oct.

Helen Yannakoudakis and Ronan Cummins. 2015. Evaluating the performance of Automated Text Scoring systems. In *Proceedings of the 10th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 213–223.

Klaus Zechner, Derrick Higgins, Xiaoming Xi, and David M. Williamson. 2009. Automatic scoring of non-native spontaneous speech in tests of spoken English. *Speech Communication*, 51(10):883–895.