MIT Computational Law Report

Unsupervised Machine Scoring of Free Response Answers—Validated Against Law School Final Exams

David Colarusso

Published on: Sep 06, 2022

URL: https://law.mit.edu/pub/unsupervised-machine-scoring-of-free-response-answers

ABSTRACT

This paper presents a novel method for unsupervised machine scoring of short answer and essay question responses, relying solely on a sufficiently large set of responses to a common prompt, absent the need for pre-labeled sample answers—given said prompt is of a particular character. That is, for questions where “good” answers look similar, “wrong” answers are likely to be “wrong” in different ways. Consequently, when a collection of text embeddings for responses to a common prompt are placed in an appropriate feature space, the centroid of their placements can stand in for a model answer, providing a lodestar against which to measure individual responses. This paper examines the efficacy of this method and discusses potential applications.

> **Keywords:** Natural Language Processing, NLP, Unsupervised Machine Learning, Machine Learning, ML, Automated Grading, Grading, Free Response Questions, Essays, Law School Exams

Introduction

Many contemporary systems for the automated scoring of essays and short answer questions rely heavily on the application of supervised machine learning, requiring a set of pre-scored model answers (Burrows et al., 2015). When used, unsupervised methods have focused largely on the discovery of features for use in the supervised training of scoring models (e.g., the discovery of latent topics). The use of unsupervised methods has been used to group similar answers, allowing for more efficient scoring by human graders (Basu et al., 2013). However, the unsupervised component of this approach fails to produce insights into the correctness of answers and so cannot qualify as scoring. Rather, scoring is left to human graders or makes use of machine comparisons to pre-labeled sample answers (supervised learning). Consequently, current methods for the automated scoring of short answer and essay questions are poorly suited to spontaneous and idiosyncratic assessments. That is, the time saved in grading must be balanced against the time required for the training of a model. This includes tasks such as the creation of pre-labeled sample answers. This limits the utility of machine grading for novel or spontaneous assessments, such as those one might deploy in an individual section of a larger course in response to an open-ended class discussion.

This paper presents a novel method for unsupervised machine scoring of short answer and essay questions given only a sufficiently large set of responses to a properly-formed common prompt. This eliminates the need for the preparation of pre-labeled sample answers. It is the author’s hope that such a method may be leveraged to reduce the time needed to grade free response questions, promoting the increased adoption of formative assessment.

This paper will describe the particulars of this method, discuss a potential application, and examine the method’s efficacy.

Methods and Application

> *“Happy families are all alike; every unhappy family is unhappy in its own way.”* - Leo Tolstoy

The method described here depends on the insight gained from the Anna Karenina Principle. Drawing its name from the first line of Tolstoy’s Anna Karenina, this principle observes that success is often more constrained than failure (Diamond, 2005). That is, there are many more ways to fail at something than there are to succeed. Applied to the grading of free response questions, this can be restated as follows. For questions where “good” answers look similar, “wrong” answers are likely to be “wrong” in different ways.

Consequently, there may be information about the correct answer to a question contained in the relative similarities between answers to that question. That is, those answers most like *the average of the answers* are likely to be “better” answers. If we are able to map answers into an appropriate feature space, we might expect that a ranking of answers based on their distance from the *average of the answers* would correspond to a ranking based on their quality.

The ability to gain actionable information from a population of answers by measuring this distance, however, is dependent on the nature of the question and its accompanying population of answers. The question must be one where “good” answers look similar, and the population must be both sufficiently large and sufficiently diverse for there to be a variety of wrong answers alongside a sizable number of correct answers.

The ability to gain actionable information from a population of answers in this way is also dependent on our ability to encode, in a computable fashion, the relevant content contained within an answer.

At present, a prompt asking respondents to “write an evocative short story” is poorly suited for the method described here. For example, “good” responses are not likely to be similar in a manner amenable to current methods for the automatic encoding of texts as numbers. Methods such as word2vec though able to capture some shadow of the responses’ semantic content will fail to capture affective content, and “good” responses are likely to differ a great deal in the former even if they are similar when it comes to the latter.

Understanding these constraints, one can construct a method for the automated unsupervised scoring of free response answers. Such a method must begin with an appropriately framed question. That is a question where all of the “good” responses look alike as measured across some category or categories amenable to numerical encoding. From here the method can be described generally as:

> 1. Produce an embedding for each answer that captures as much of the relevant information as possible.
>
> 2. Find the centroid for all of the embeddings in your population of answers and impute the location of a “correct” model answer.

> 3. Measure the distance between each answer's embedding and the "correct" answer (e.g., the answers' centroid or medoid).

Assuming the constraints above, the distance of an answer's embedding from the centroid serves as a proxy for its quality. The closer an answer is to the centroid, the "better" it is. It may, however, be desirable to measure the distance from a nearby location. For example, if it is decided that the "best answer" should be *the best answer to appear in an actual answer*. In this case, the embedding nearest the centroid (the medoid) could be used as the point against which others were measured.

If one wishes, they can expand upon this simple ordering to produce traditional grade markings. To do this:

> 4. Convert the answers' distances from the model answer into z-scores for the population of answers.
>
> 5. Translate these z-scores into some known grading scale.

Though there is no generally agreed upon translation between z-scores and traditional letter grades, it is customary for individual graders or institutions to settle upon particular grading curves. That is, the mean score may be set to a B-, corresponding to a z-score of X, and so on. Translations could also be made to other grading scales, such as pass-fail, where a pass is a z-score over some value Y.

The code for the above method, including sample translations between z-scores and traditional letter grades can be found at https://github.com/colarusso/free-response-sc oring. This implementation stands upon a foundation of prior work, including the Python programming language, various mathematics, statistics, analytics, and NLP tools—Pandas, NumPy, SciPy, scikit-learn, Pingouin, and SpaCy (Van Rossum & Drake, 1995; McKinney et al., 2010; Harris et al., 2020; Virtanen et al., 2020; Pedregosa et al., 2011; Vallat 2018; and Honnibal & Montani, 2017).

The implementation encoded at the above link allows for the production of embeddings based on a selection of popular language models, including word2vec, DistilBERT, and RoBERTa. The feature space used to represent an answer, and against which one calculates distance, could include additional features such as reading level and other easily computed measures. The inclusion of any features, however, should be dependent on their ability to capture information relevant to an answer's quality. Additionally, their contribution should be weighted based on their relative importance.

As discussed above, the ability to encode information relevant to an answer's quality is necessary for the successful application of this method. An inability to capture the entirety of this information will lead to predictable errors. For example, a language model such as word2vec may fail to register the similarity between two texts expressing the same idea when one of the two makes use of idioms (Mikolov, 2013). One might call this class of problems *the elephant in the room*. Consequently, we can expect that output of the described method will predictably fail in such situations. This calls to mind the George Box quote observing that "all models are wrong, but some are useful." From this observation, we can draw two actionable insights.

> - Because models are wrong, their output should start, not end, discussion.
>
> - To determine if a model is useful, one must ask "compared to what?"

These insights suggest features one should apply when using the method as well as a means for evaluating its efficacy.

Though the method described above is capable of producing scorings for individual answers in a population of answers, the known limitations of such suggest that these markings should not be the final word. Rather, while such limitations persist, it is prudent to present these markings for evaluation by a human grader before they are finalized. Therefore, one can expand upon the method by adding the following steps.

> 6. Order the answers according to their markings.
>
> 7. Present this ordered list to a human grader or graders for review.

The nature of the presentation should be designed to provide the human grader with a grading experience superior to current practice. One such presentation may be one where the graders are able to easily reorder the ordered list of machine-scored answers. Ideally, this reordering would be accompanied by an automatic rescoring of the relevant text(s). For example, a grader could be presented with a screen consisting of multiple columns corresponding to grade markings (e.g., A, B, C, D, and F) with answers occupying the columns based on their machine score. Graders could then move answers up or down within a column or across columns such that their order represented their relative quality. A text's position in this ordering would then determine its score. Alternatively, graders could enter a score directly, absent the need to manipulate an answer's position, prompting an automated reordering of the list as needed.

Answers for which the feature space fails to capture relevant content would likely present as outliers (e.g., those subject to *the elephant in the room* problem discussed above). That is, they would likely receive poor marks and exist at the far end of the grade distribution. This suggests that outliers be watched carefully so that they may be moved manually by a person capable of evaluating the relevant content.

Additionally, there is no hard constraint on the grouping of answers into traditional letter grades. The ordering presented to a grader could divide answers into alternative groupings, e.g., *pass* and *fail*. Such a division would likely mark the overwhelming majority of answers as *pass*, allowing graders to quickly review and assess the scorings for the minority of answers marked *fail*.

Given that the method described takes an answer to be part of a population of answers and that it is within the context of this population that final human grading takes place, it seems appropriate to evaluate the method's efficacy at this level. Consequently, the method will be considered effective to the extent that it is capable of producing an ordered list of answers which resembles the ordered list a human grader would produce.

To test the efficacy of the above method, more than one thousand student answers to a set of thirteen free response questions, drawn from six Suffolk University Law School final exams, taught by five instructors, were run through the algorithm described above, producing ordered lists of machine-marked answers. This corresponds to steps one through four above as these answers were graded by human instructors before being run through the method described. That is, the grader could not have been influenced by the machine scores. These responses and their associated grade data were acquired and this research conducted after approval of Suffolk University's Office of Research and Sponsored Programs (ORSP) which oversees all human subject research at Suffolk University. These thirteen questions constitute all but one of the essay/free response questions present on the six exams obtained. The one excluded question was excluded because it was open-ended and did not ask for a single correct answer. Rather, it asked students to choose and discuss a subset of cases covered in the course. Consequently, "good" answers would not necessarily have looked similar.

Again, the code used to conduct the scoring described here is available at https://github.com/colarusso/free-response-scoring. One can also find three of the six exams at this link. The results described here made use of the code's default parameters (i.e., a word2vec language model with the model answer set at the centroid). Each set of answers was run through the method 100,000 times and the output lists were compared to the ordering obtained by the students' actual grades.

To avoid overfitting, the code used was originally tested and tuned on free response answers from one of the author's classes, and these texts were not included in the set of evaluation texts discussed below.

Free response answers were chosen for inclusion based on their character. That is, they had to ask for "good" answers that looked similar. Open-ended questions designed to elicit a constellation of dissimilar "good" answers (analogous to the short story example above) were excluded. This resulted in the exclusion of the question asking students to choose and discuss a subset of cases.

Results

To determine how similar a list's order was to that produced by a human grader, the lowest number of neighbor swaps needed to transform the ordering of a list into that of the human ordering was calculated (Gupta, 2021). That is, when two adjacent exams in an ordered list change positions this movement is called a swap, and it is only through swaps that exams can be reordered. Consequently, the minimum number of swaps needed to reorder a list so that it matches that produced by the human grader provides information on the quality of the initial ordering.

This measure was chosen primarily for its similarity to the task of reordering asked of human graders in the application envisioned above. That is, a perfect machine ordering would not require a human grader to rearrange any of the answers while a lower quality ordering would require many swaps as a grader moved texts up or down the ordering.

After converting both human and machine markings into z-scores, the average number of swaps needed to transform an ordering into that of the human grader were found for the machine ordering and the pseudo-random shuffles across 100,000 runs of each exam question. The averages of the swaps needed were then compared. A summary of the results can be seen in **Table 1.**

For the data presented in **Table 1**, the Cohen's d for the number of swaps needed between the average pseudo-random ordering and the average machine ordering is 1.03, which is large (Cohen, 1988). The p-value for a paired t-test of the two populations' swaps, with the pseudo-random group acting as the untreated group and the machine-grader acting as treatment, came to 0.000000334, allowing us to reject the null hypothesis that the machine's ordering is equivalent to a random shuffle.

Consequently, these results suggest that the method described is capable of conducting an unsupervised scoring of free response answers across a variety of course sizes and instructors. It is by no means a perfect scoring, but it is hoped that it may be sufficient to improve the process of grading free response questions through application of the methodology described above, additional feature engineering, and further parameter tuning.

For readers interested in how well the method agreed with human graders, the mean intraclass correlation, ICC(3,1), for the method paired with a human grader was 0.38, and the median was 0.41. See **Table 1**. The mean and median quadratically weighted Cohen's kappa for rounded z-scores came to 0.40 and 0.45 respectively. To address the fact Cohen's kappa expects ordanal markings, z-scores were multiplied by 10 and rounded to the nearest integer to produce ten categories per standard deviation. See **Table 1**. The observed performance straddles the boundary between *fair* and *moderate* agreement (Landis & Koch, 1977). It is, however, arguably within the range of agreements found for human graders of essay questions (see e.g., Gugiu et al. 2012, summarizing past reported human agreements). It is worth noting that the literature on agreement between human graders of free response questions shows a good deal of variability, with both high and low agreements. Consequently, it is not difficult for our method to find itself in this range. Consider, for example, the conclusion reached by G.C. Bull who found that the random assignment of grades would be almost as helpful as those applied by human graders (Bull, 1965).

The performance of the method was examined across a number of different parameters, including the means of vectorization (i.e., word2Vec, DistilBERT, and RoBERTa) and scoring scales with varying granularity (e.g., continuous and categorical). Results from such runs can be found along with code and data at: https://github.com/colarusso/free-response-sc oring. These permutations resulted in statistically significant differences between the number of swaps needed to transform pseudo-random and machine orderings into that of the human graders.

Future Work

More work is needed to see how such a method's performance varies given different population sizes and feature spaces. There is likely a good deal of performance to be gained through additional feature selection and parameter tuning. The application of the method as an aid in the human grading process should be evaluated explicitly. That is, answers should be run through the entirety of the seven steps described above. This evaluation should look to see to what extent the method described reduces the time needed for grading, to what extent the presentation of pre-ordered lists biases graders, and under what conditions such bias would be acceptable. That is, there is the possibility that graders, presented with pre-ordering, will simply adopt the machine's ordering, failing to act as an independent check. Depending on the use case, including the sensitivity required of markings and the cost of miscategorogrization, this may or may not be acceptable.

Given that the current incarnation of this method achieves performance approaching that of the worst human graders, the author believes it would be irresponsible for it to replace human graders at this time. Rather, they suggest its use as a cognitive exoskeleton to help balance the load of grading, allowing human graders to do more (e.g., the adoption of more formative assessment).

Acknowledgments

I would like to thank the Editor and the Chief, Kathleen and Charles Colarusso, for a life-long reservoir of support and feedback, as well as Jessica Colarusso for her partnership in all things.

Thanks is also owed to professors Sarah Boonin, Erin Braatz, John Infranca, Jeffrey Lipshaw, and Sharmila Murthy for providing access to historic exam data.

References

Basu, Sumit & Jacobs, Chuck & Vanderwende, Lucy. (2013). Powergrading: a Clustering Approach to Amplify Human Effort for Short Answer Grading. TACL. 1. 391-402. 10.1162/tacl_a_00236.

Burrows, S., Gurevych, I. & Stein, B. The Eras and Trends of Automatic Short Answer Grading. *Int J Artif Intell Educ* **25,** 60–117 (2015). https://doi.org/10.1007/s40593-014-0026-8

Bull, G. M. (1956). An Examination of the Final Examination in Medicine. The Lancet, 271, 368-372.

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences*. Routledge. ISBN: 9780805802832 (2nd ed., pp. 26027)

Diamond, Jared M. (2005). Guns, germs, and steel : the fates of human societies. New York :Norton, Gugiu, M. R., Gugiu, P. C., & Baldus, R. (2012). Utilizing Generalizability Theory to Investigate the Reliability of Grades Assigned to Undergraduate Research Papers. Journal of MultiDisciplinary Evaluation, 8(19), 26–40.

Gupta, Shivam (2016, November 25). Number of swaps to sort when only adjacent swapping allowed. GeeksforGeeks. https://www.geeksforgeeks.org/number-swa ps-sort-adjacent-swapping-allowed/

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Cournapeau, Wieser, E., Taylor, J., Berg, S., Smith, N., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M., Brett, M., Haldane, A., Fernández del Río, J., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., & Oliphant, T. (2020). Array programming with NumPy. *Nature*, 585(7825), 357–362. https://doi.org/10.1038/s41586-020-2649-2

Honnibal, Matthew and Montani, Ines. spacy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing. *To appear*, 7(1), 2017.

Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B., Bussonnier, M., Frederic, J., Kelley, K., Hamrick, J., Grout, J., Corlay, S., Ivanov, P., Avila, D., Abdalla, S., Willing, C., & Jupyter development team. (2016). Jupyter Notebooks – a publishing format for reproducible computational workflows (F. Loizides & B. Scmidt, Eds.; pp. 87–90). *IOS Press*. https://doi.org/10.3233/978-1-61499-649-1- 87

Landis, J. R., & Koch, G. G. (1977). The Measurement of Observer Agreement for Categorical Data. Biometrics, 33(1), 159-174.

McKinney, W. (2010). Data Structures for Statistical Computing in Python. *Proceedings of the 9th Python in Science Conference*, 56–61. https://doi.org/10.25080/Majora-92bf1922-00 a

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. Advances in Neural Information Processing Systems, 26. https://proceedings.neurips.cc/paper/2013/hash/9aa42b31882ec039965f3c4923ce901b-Abstract.html

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesna, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12(Oct), 2825–2830.

Vallat, R. (2018). Pingouin: Statistics in Python. Journal of Open Source Software, 3(31), 1026. https://doi.org/10.21105/joss.01026

Van Rossum, G., & Drake Jr, F. L. (1995). *Python reference manual*. Centrum voor Wiskunde en Informatica Amsterdam.

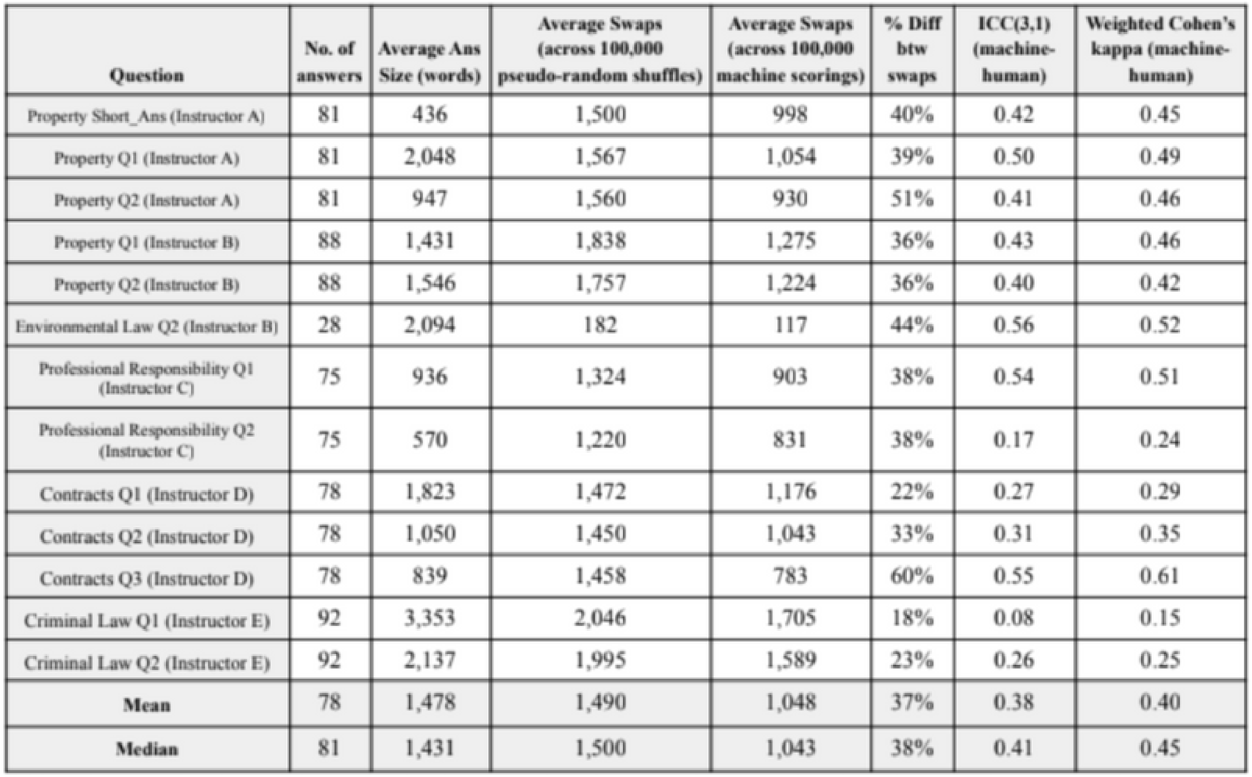
Virtanen, P., Gommers, R., Oliphant, T., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S., Brett, M., Wilson, J., Millman, K., Mayorov, N., Nelson, A., Jones, E., Kern, R., Larson, E., Carey, CJ, Polat, I., Feng, Y., Moore, E., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E.A., Harris, C., Archibald, A., Ribeiro, A., Pedregosa, F., van Mulbregt, P., SciPy 1.0 Contributors. (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17(3), 261-272. https://doi.org/10.1038/s41592-019-0686-2

Competing Interest

The methods described here are the subject of USPTO Patent Application Serial No. 17/246,563 which the author has assigned to Suffolk University under an agreement to share any potential revenue.

Supplemental Materials

An implementation of the above methods, a more detailed description of data cleaning, and questions from three of the six exams can be found on GitHub at: https://github.com/colarusso/free-response-scoring



Question	No. of answers	Average Ans Size (words)	Average Swaps (across 100,000 pseudo-random shuffles)	Average Swaps (across 100,000 machine scorings)	% Diff btw swaps	ICC(3,1) (machine-human)	Weighted Cohen's kappa (machine-human)
Property Short_Ans (Instructor A)	81	436	1,500	998	40%	0.42	0.45
Property Q1 (Instructor A)	81	2,048	1,567	1,054	39%	0.50	0.49
Property Q2 (Instructor A)	81	947	1,560	930	51%	0.41	0.46
Property Q1 (Instructor B)	88	1,431	1,838	1,275	36%	0.43	0.46
Property Q2 (Instructor B)	88	1,546	1,757	1,224	36%	0.40	0.42
Environmental Law Q2 (Instructor B)	28	2,094	182	117	44%	0.56	0.52
Professional Responsibility Q1 (Instructor C)	75	936	1,324	903	38%	0.54	0.51
Professional Responsibility Q2 (Instructor C)	75	570	1,220	831	38%	0.17	0.24
Contracts Q1 (Instructor D)	78	1,823	1,472	1,176	22%	0.27	0.29
Contracts Q2 (Instructor D)	78	1,050	1,450	1,043	33%	0.31	0.35
Contracts Q3 (Instructor D)	78	839	1,458	783	60%	0.55	0.61
Criminal Law Q1 (Instructor E)	92	3,353	2,046	1,705	18%	0.08	0.15
Criminal Law Q2 (Instructor E)	92	2,137	1,995	1,589	23%	0.26	0.25
Mean	78	1,478	1,490	1,048	37%	0.38	0.40
Median	81	1,431	1,500	1,043	38%	0.41	0.45

Table 1. Summary data for graded texts showing the: (1) number of answers; (2) average size of texts in words; (3) average number of swaps for 100,000 pseudo-random orderings; (4) the average number of swaps for 100,000 machine orderings; (5) the percent difference between the two average swaps; (6) ICC(3,1)

between the human and machine z-scores; and (7) quadratically weighted Cohen’s kappa between the human and machine scores where the z-score were first multiplied by ten and rounded to the nearest integer to convert continuous into ordinal data.