# Using Machine Learning Algorithms to Categorize Free Responses to Calculus Questions[1]

Matthew Thomas
Ithaca College

Spencer Bagley
University of Northern Colorado

Mark Urban-Lurain
Michigan State University

*Researchers in various science disciplines have begun exploring use of machine learning algorithms to categorize students' answers to constructed-response tasks, achieving inter-rater reliability on par with that between expert raters. We report on a proof-of-concept experiment in which we categorized student responses to conceptually-focused tasks on a calculus final exam. Our results were only modestly successful, but promising. We identify ways in which responses to mathematics tasks are uniquely challenging for these algorithms, and ways in which the algorithms' performance on mathematics tasks can be improved.*

*Keywords:* assessment, calculus, machine learning, constructed response

Advances in machine learning algorithms have introduced the possibility of using computers to categorize written responses to questions based on linguistic patterns. When provided a corpus of hand-scored responses, these programs are able to identify patterns in the responses, and are able to automatically evaluate future responses based on similarities to responses already scored. This has been demonstrated to be successful in several science disciplines and statistics, but these techniques have not yet been applied to mathematics courses more broadly.

Application of these algorithms to education research and to the classroom has many potential benefits. In the classroom, these tools may allow for automated categorization of responses to open questions, so that students in large classes (e.g., large lectures or MOOCs) can receive immediate feedback, as they might in a system such as WeBWorK, but on open-ended, conceptually-focused questions. For research, these algorithms have the potential to identify both students' ways of thinking and how they are connected.

Prior efforts at measuring conceptual understanding have mainly relied on multiple-choice-based concept inventories (see, e.g., Libarkin, 2008); specifically, Epstein (2013) developed a concept inventory for calculus. However, the multiple choice format of a concept inventory is inherently restrictive, and prior research has identified problems with the CCI (Gleason, White, Thomas, Bagley, & Rice, 2015). Machine learning algorithms may allow for the creation of new instructional and assessment tools which capture nuances that concept inventories cannot.

In this study, we consider a proof-of-concept experiment in which we gathered student answers to free-response questions on a calculus final exam and attempted to use these algorithms to analyze the data. In doing so, we identify unique challenges to using these methods in mathematics, and provide ideas for how these challenges can be overcome.

## Background

Our motivation for conducting this study is to improve assessment of conceptual understanding in calculus. Our understanding of assessment is informed by the National Research Council's (NRC, 2001) model: cognition, or models of student understanding;

[1] Acknowledgements: This material is based upon work supported by the National Science Foundation (DUE grants: 1438739, 1323162, 1347740, 0736952 and 1022653). Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the NSF.

observations, or the tasks by which students' understanding is elicited in an observable form; and interpretation, in which the assessor uses their cognitive models to make sense of students' observed behavior. In particular, in order to make valid inferences, the tasks must be appropriate to elicit the desired types of cognition, especially when measuring conceptual understanding.

Most tasks targeting conceptual understanding are either forced-response tasks (e.g., multiple-choice or true-false questions, where students must select the correct answer from a prescribed list of possibilities) or constructed-response tasks (where students must explain concepts in their own words). While both item types can serve to elicit students' conceptual understanding, students' own writing on open-ended questions reveals more about their conceptions and misconceptions than does their performance on a multiple-choice exam (Birenbaum & Tatsuoka, 1987), as constrained-response questions obscure nuances and partial conceptions in student thinking (Hubbard, Potts, & Couch, 2017).

Our work is informed by the Automated Analysis of Constructed Response (AACR) project (https://msu.edu/~aacr/), an ongoing NSF-funded project which seeks to use machine learning to rapidly and efficiently categorize students' responses to open-ended conceptual questions. First, researchers develop and administer constructed response questions to students, and experts categorize the student responses. Then, the coded data are given to a machine-learning algorithm, which builds a model of expert rating using a subset of the coded data. The model's performance is assessed through randomized crossvalidation, calculating a measure of inter-rater reliability (usually Cohen's kappa). Further, several different models built with different machine learning techniques can be combined into an ensemble model, weighting each model by its confidence level; the version of the algorithm we used is an ensemble of 8 separate algorithms.

AACR has achieved impressive performance in several disciplines, including biology (Beggrow, Ha, Nehm, Pearl, & Boone, 2014; Prevost, Smith, & Knight, 2016), chemistry (Haudek, Prevost, Moscarella, Merrill, & Urban-Lurain, 2012), interdisciplinary understandings of energy (Park, Haudek, & Urban-Lurain, 2015), and even statistics (Kaplan, Haudek, Ha, Rogness, & Fisher, 2014). This evidence suggests the AACR approach may be successful in other mathematics classes, though there may be unique obstacles not faced in other disciplines.

## Methods

A total of 67 students in two sections of a coordinated introductory calculus course at a mid-sized university in the Rocky Mountain region of the United States were given the following questions on a common final exam:

1. The limit definition of the derivative of a generic function f(x) is: $\lim_{h \to 0} \frac{f(x+h)-f(x)}{h}$.
    a. What does the numerator mean?
    b. What does the denominator mean?
    c. Why are we taking the limit as h approaches 0?
    d. Explain why the limit definition given above aligns with the overall meaning of the derivative.
2. Suppose the derivative of a function f(x) is negative everywhere on the interval x = 2 to x = 3. Where on this interval (i.e. for what x-value) does the function f have its maximum value? Carefully explain how you know your answer is correct.

We chose to examine student responses to these two questions for several reasons. First, these are conceptually-focused questions, and our ultimate aims are to move toward the development

of a library of computer-gradeable items assessing students' conceptual understanding of calculus. Second, student responses to these two questions tended to contain more words than symbols or calculations, which makes the data they produce more amenable to analysis using these machine learning techniques. Third, these questions were designed to elicit a wide variety of ways of interpreting and thinking about the derivative concept.

The two instructors, who are not authors on this report, provided us with anonymized data in the form of digital scans of students' responses to these two questions. Not every student answered every part of both questions; see Table 1 for specific values of N for each question.

We transcribed students' handwritten responses into machine-readable text. In particular, we rendered mathematical symbols into words (for example, we rendered " → " as "to") and corrected spelling errors (for example, we corrected various misspellings of "infinitely").

The first two authors independently coded student responses using an emergent open coding process. We independently examined all the student responses to each part of the two questions to identify recurring themes and regularities in student responses, and used these themes and regularities to produce coding schemes. We then met to compare categories, resolve discrepancies, and produce a consensus coding scheme. In the final consensus coding scheme, each part of the two questions had four to six non-exclusive categories (or bins) of student responses, and we coded each student response as either belonging or not belonging to each.

After this coding process was complete, the third author used the coded data as input for the AACR algorithms. The first two authors then performed an error analysis by reviewing the output of the algorithms, looking for bins that performed particularly well or particularly poorly, then seeking to discover possible reasons for the performance of each bin.

## Results

The emergent open coding process resulted in the creation of categories for each question. While space constraints preclude us from giving a full description of every bin of student responses, categories included features of responses such as whether the student identified a "delta" or change in particular values, used common arguments such as "the derivative is negative, so the function is decreasing," or highlighted a geometric or graphical interpretation of a piece of the difference quotient.

Overall, our efforts were only modestly successful, but promising. Our main measure of how well the algorithm performed is inter-rater reliability, as measured by Cohen's kappa. Our kappas ranged from 0.759 to -0.09 (see Table 1). According to Landis and Koch (1977), kappa values between 0.61 and 0.8 represent "substantial" agreement; values between 0.41 and 0.6 represent "moderate" agreement; between 0.21 and 0.4, "fair" agreement; between 0 and 0.2, "slight" agreement; and below 0, "poor" agreement.

| Question | N | Kappas for each category (largest to smallest) | | | | | |
|---|---|---|---|---|---|---|---|
| Question 1a | 67 | 0.616 | 0.576 | 0.290 | 0.000 | | |
| Question 1b | 67 | 0.546 | 0.546 | 0.533 | 0.000 | | |
| Question 1c | 66 | 0.746 | 0.579 | 0.488 | 0.417 | 0.159 | 0.000 |
| Question 1d | 65 | 0.690 | 0.278 | 0.177 | 0.000 | 0.000 | -0.090 |
| Question 2 | 67 | 0.759 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |

Table 1: Summary of inter-rater reliability between expert coding and algorithm coding

To gain more insight into the practical performance of the algorithm, we conducted an error analysis, carefully examining all the false positives and false negatives. We looked in particular for commonalities in misclassified responses, as they might reveal the specific difficulties the algorithm faced when attempting to classify responses to math tasks, and thus suggest ways to improve the performance of the algorithm. We illustrate our analysis with a few examples.

**Homogeneity in Responses**

While conducting this investigation, we discovered some elements of these algorithms that worked well in this setting and others which did not. When responses that we coded into a single category all contained the same key phrase (or only small variations), the algorithm tended to be particularly successful in matching our coding. For example, in question 2, one bin included noticing that the derivative was negative. There were very few phrasings students used to capture this idea (for example, "since the derivative of f(x) is negative that means that f(x) is decreasing"), and these phrases showed up repeatedly. In this category, the algorithm achieved a kappa value of 0.759, indicating substantial agreement with our coding. Question 1c included a category about estimating the slope, which performed well for similar reasons; most responses in this bin used the phrase "slope of the tangent line" or the word "secant." Here, the kappa value was 0.579. In categories like these, the algorithm could more easily pick out a pattern that characterizes all the responses in the bin, and thus was able to perform better.

On the other hand, we noted that the algorithm performed poorly on several categories in which the responses were not homogeneous enough. For example, one bin in question 1c was the "cancels out" bin; responses in this category expressed the idea that the $h$ in the definition of the derivative should not appear in the correct end result. This idea was expressed in many different ways, such as "cancel out," "get rid of," "plug 0 into $h$," "be removed," "equal zero," "eliminate," and "substitute 0 for $h$." Due to the inability of the computer to recognize these phrases as describing the same ideas, this bin had a kappa value of 0.

**Sample Size**

One clear limitation of our data set was the sample size. Machine learning algorithms require a large enough sample size (often a few hundred) so that patterns in the responses can be identified. In order for the algorithm to differentiate responses, it must detect patterns that exist within the categorized responses *and* do not exist among the remaining responses. Because there will be variation in the phrasing of ideas, enough responses need to exist in order to identify the various ways the same concept may be expressed. This was particularly clear in question 1c's category of "cancels out," as discussed above. Increasing the sample size would increase the likelihood of the algorithm learning that the many variations express the same idea.

**Bag-of-Words Model**

We also consider particular issues which may occur using these techniques in mathematics which may not occur in other subject areas. The ensemble of algorithms used by the AACR project used a bag-of-words (BOW) approach. The raw data is broken into 1-3-word n-grams (that is, words, pairs of words, and three-word phrases). When only 1-word n-grams are used,

there is no sense of order or proximity, only the collection of words in a response. When 3-word n-grams are used, there is some sense of proximity, but only within a distance of 3 words.

For many responses to our tasks, the only substantive difference between examples and non-examples of a category was the order of the words they used. One category for question 2 was discussing the derivative being negative. One of the false positives in this category was this response: "If the function is negative on the interval $x = 2$ to $x = 3$ because of the derivative, then the function's maximum value is at $x = 2$. If the function is negative then it means that the slope is decreasing causing the maximum value between 2 and 3 to be at $x = 2$." Compare this response to a true positive: "The maximum is located on $x = 2$. Since the derivative is negative at every point on the interval, the function's slope is known to be negative for the entirety of the interval. Since $f(x)$ is always decreasing on the interval, the leftmost point is the maximum." The false positive response contains many of the same words as the true positive response (e.g., "derivative," "negative," "slope," "maximum," "decreasing"), but uses them in a different order. A strictly-BOW model cannot easily detect the difference between these two responses.

We suspect that the order of words matters more in mathematical writing than it does in writing in other STEM disciplines. Writing a correct description of a mathematical procedure, or rendering mathematical notation into text, likely requires comparatively more precise attention to word order; in contrast, a correct description of the principles underlying evolution likely depends more on the word choice than the word order.

## Future Directions

While we haven't achieved inter-rater reliability on par with that between human experts, these results provide promise that these machine learning algorithms may be applied to mathematics, while also indicating some ways in which these methods may need to be modified. We aim to begin with a larger sample size to determine whether some of the challenges are solely due to the sample size or are connected with the language of mathematics. We also aim to iteratively refine our coding scheme to allow for a more even split of the responses to aid in the pattern recognition. Additionally, we may need to consider details of how the input is parsed by the software, in particular by considering how to best represent mathematical notation not easily translated into words, such as arrows or functional notation. For instance, our initial exploration with another text classification tool called LightSIDE suggests that we may achieve better performance by replacing the character $\Delta$ with the word "delta."

We also aim to develop more, and better, questions with which to gather data. We initially thought question 1 would be a good choice, because it elicits student understanding of the limit definition of the derivative and tends to produce "wordy" responses. However, our analysis suggests that the question is problematic, since it asks students to make meaning of mathematical notation, leading to responses containing notation or direct translations of notation into words. We hope to create questions whose answers would be more descriptive than notational.

## Audience Questions

- What questions might elicit more descriptive language in student responses?
- The more data we have, the better the algorithms will perform. Are there existing repositories of large numbers of text-based student responses to conceptual questions?
- We want to make our models useful for instructors and researchers. What common understandings exist about the concepts that are most important for students?

## References


Birenbaum, M., & Tatsuoka, K. K. (1987). Open-ended versus multiple-choice response formats—it does make a difference for diagnostic purposes. *Applied Psychological Measurement, 11*(4), 385-395.

Beggrow, E. P., Ha, M., Nehm, R. H., Pearl, D., & Boone, W. J. (2014). Assessing scientific practices using machine-learning methods: How closely do they match clinical interview performance? *Journal of Science Education and Technology, 23*(1), 160-182. https://doi.org/10.1007/s10956-013-9461-9

Epstein, J. (2013). The calculus concept inventory—measurement of the effect of teaching methodology in mathematics. *Notices of the American Mathematical Society, 60*(8), 1018-1026. https://doi.org/10.1090/noti1033

Gleason, J., White, D., Thomas, M., Bagley, S., & Rice, L. (2015). The calculus concept inventory: A psychometric analysis and framework for a new instrument. In T. Fukawa-Connelly, N. E. Infante, K. Keene, & M. Zandieh (Eds.), *Proceedings of the 18th Annual Conference on Research in Undergraduate Mathematics Education* (pp. 135–149).

Haudek, K. C., Prevost, L. B., Moscarella, R. A., Merrill, J., & Urban-Lurain, M. (2012). What are they thinking? Automated analysis of student writing about acid–base chemistry in introductory biology. *CBE-Life Sciences Education, 11*(3), 283-293. https://doi.org/10.1187/cbe.11-08-0084

Hubbard, J. K., Potts, M. A., & Couch, B. A. (2017). How question types reveal student thinking: an experimental comparison of multiple-true-false and free-response formats. *CBE-Life Sciences Education, 16*(2), ar26. https://doi.org/10.1187/cbe.16-12-0339

Kaplan, J. J., Haudek, K. C., Ha, M., Rogness, N., & Fisher, D. G. (2014). Using lexical analysis software to assess student writing in statistics. *Technology Innovations in Statistics Education, 8*(1). Retrieved from https://escholarship.org/uc/item/57r90703

Landis, J. R., & Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics, 33*(1), 159-174. https://doi.org/10.2307/2529310

Libarkin, J. (2008). Concept inventories in higher education science. In National Research Council Promising Practices in Undergraduate STEM Education Workshop 2 (pp. 1–13). Washington, D. C. Retrieved from http://www7.nationalacademies.org/bose/Libarkin_CommissionedPaper.pdf

National Research Council. (2001). *Knowing what students know: The science and design of educational assessment*. National Academies Press.

Park, M., Haudek, K., & Urban-Lurain, M. (2015). Computerized lexical analysis of students' written responses for diagnosing conceptual understanding of energy. In *National Association for Research in Science Teaching (NARST) 2015 Annual International Conference*. Retrieved from http://create4stem.msu.edu/publication/3361

Prevost, L. B., Smith, M. K., & Knight, J. K. (2016). Using student writing and lexical analysis to reveal student thinking about the role of stop codons in the central dogma. *CBE-Life Sciences Education, 15*(4), ar65. https://doi.org/10.1187/cbe.15-12-0267