Machine learning for automated content analysis: characteristics of training data impact reliability

Rebeckah Fussell, Ali Mazrui, and N. G. Holmes
Laboratory of Atomic and Solid State Physics, Cornell University, Ithaca, New York 14853, USA

Natural language processing (NLP) has the capacity to increase the scale and efficiency of content analysis in Physics Education Research. One promise of this approach is the possibility of implementing coding schemes on large data sets taken from diverse contexts. Applying NLP has two main challenges, however. First, a large initial human-coded data set is needed for training, though it is not immediately clear how much training data are needed. Second, if new data are taken from a different context from the training data, automated coding may be impacted in unpredictable ways. In this study, we investigate the conditions necessary to address these two challenges for a survey question that probes students' perspectives on the reliability of physics experimental results. We use neural networks in conjunction with Bag of Words embedding to perform automated coding of student responses for two binary codes, meaning each code is either present or absent in a response. We find that i) substantial agreement is consistently achieved for our data when the training set exceeds 600 responses, with 80-100 responses containing each code and ii) it is possible to perform automated coding using training data from a disparate context, but variation in code frequencies (outcome balances) across specific contexts can affect the reliability of coding. We offer suggestions for best practices in automated coding. Other smaller-scale investigations across a diverse range of coding scheme types and data contexts are needed to develop generalized principles.

2022 PERC Proceedings edited by Frank, Jones, and Ryan; Peer-reviewed, doi.org/10.1119/perc.2022.pr.Fussell

I. INTRODUCTION

Content analysis is crucial in Physics Education Research (PER) for interpreting open-ended data generated by students [1]. The scale of these studies, however, is often limited to a few hundred students at a single institution because it is incredibly time consuming to apply a coding scheme. As techniques in machine learning and Natural Language Processing (NLP) advance, education researchers see many promising potential applications to content analysis. Integrating machine learning and NLP with traditional content analysis methods could lead to dramatic improvements in the scale at which content analysis can be applied. This improved scaling would allow researchers to analyze all at once large data sets collected from students at a variety of universities, across a wide range of years, and across a wide variety of student populations and conditions.

Previous science education research that incorporates NLP into content analysis falls into two categories of machine learning methods: supervised and unsupervised. Unsupervised methods organize or cluster data without considering any labels applied to the data. These methods have been used in PER, for example, to systematically study trends over time in the PERC proceedings [2], to highlight words in student writing that are likely to be associated with a code [3], and to develop a construct map that identifies patterns in student writing [4]. Unsupervised approaches can aid qualitative researchers in noticing patterns in their data but cannot be used to apply an a priori coding scheme.

Supervised methods, on the other hand, can use a coding scheme and a bank of coded data to automate coding of new data (such as in [5]). While the coding scheme can be developed using unsupervised methods (as seen in Ref. [4]), here we focus on supervised methods where one trains an algorithm to replicate a coding scheme from coded training data alone.

The primary downside of using supervised methods is that their success generally scales with the amount of training data provided. Generating the training data, particularly via an established coding scheme, requires a significant upfront human coding investment. Studies that have used this approach have often used small amounts of data ($N = 67$ and $N \approx 150$) and found mixed results: across categories and question types within the same coding scheme, Cohen's kappa (a measure of inter-rater reliability) spans the full range from 0 to 0.9 [6, 7]. Other methods using coding schemes where human inter-rater reliability falls in the 0.6 - 0.75 range have achieved inter-rater reliability with the original coder on par with another human [5, 8]. When using machine learning to analyze education data, there is rarely an abundance of coded data due to limited resources in data collection and human coding time. Furthermore, it is hard to know how much coded data are necessary to achieve particular consistent levels of reliability in a large-scale study.

Educational data sets are also likely to contain outcome imbalances where, for example, the frequencies of codes across all student responses may be very different for different codes or populations of students. They may also contain feature imbalances where, for example, a small subset of the responses that contain a code might express that idea quite differently. Coding performed by machine learning algorithms can be systematically biased by both outcome imbalances and feature imbalances [9].

Thus, before NLP can be applied to large-scale studies, it is crucial to test simpler and smaller data sets to understand what characteristics of coded PER data are sufficient to get reasonably reliable machine coding. First, researchers need to know how much human coding time investment they need to make before an algorithm can take over. Second, they need to understand just how different new data can be from the training data while still being able to trust the machine coding. The answers are likely to depend deeply on the complexity of the coding scheme, the type of student data being analyzed, and the different characteristics of various student sub-populations. Researchers need a sense of what to expect before investing huge amounts of time in coding a training set. Thus generalized principles should be gleaned from a bank of smaller-scale tests across a variety of contexts.

In this study, therefore, we test data from an open-response survey question that probes students' perspectives on measurement reliability [10]. The question was used across three years with students at one university. We ask the following research questions: 1) How much human-coded data are needed in the training data set to achieve substantial agreement with human coders? 2) How do the characteristics of the training set, including the prevalence of a code and the context of the responses, affect the reliability of machine coding?

II. METHODS

A. Data Sources

We selected methods that were suitable for a large-scale study that spans multiple types of student populations. Thus, for our study we used neural networks where accuracy increases as training set size increases. Furthermore, we selected a survey question that has been asked previously at multiple institutions to gain insight into students' perspectives on experimental results: "How do you know whether or not an experimental result is acceptable or trustworthy? What gives you confidence that the data is trustworthy?" [10].

The question was posed to students enrolled in an introductory physics lab course at Cornell University. This lab course prioritizes students learning experimental skills and developing expert-like mindsets toward experimental physics. Data were collected in the spring semesters of 2019, 2021, and 2022 and divided into three subsets based on when and in what form the question was administered to students (see Table I). All responses in subsets I & III are from students in their first semester of a skills-based experimental physics course, while subset II contains a mix of students in the first and second semester of the lab sequence.

TABLE I: Data subsets used in the study.

Subset	Set type	How asked?	Year	N
I	Train	Pre homework	2019	376
II	Train	Post survey	2021	361
III	Test	Pre survey	2022	452

In Subset I, the survey question was assigned to students as a homework question in the first week of class. In Subsets II & III, the survey question was posed to students in a Pre- or Post-class survey. This resulted in significant differences in the typical writing style between subset I and subsets II & III as students tended to put more effort into the question when posed as homework. The average length of a response in subset I was 48 words while the average length of responses in subsets II & III was 21 and 24, respectively.

B. Coding Scheme

The authors iteratively revised the coding scheme from [10] using the new data set and developed a new 7-code scheme. The codes are not mutually exclusive because a single response can contain multiple ideas. Thus, responses are coded one code at a time: the coder reads a response looking for any ideas that match the inclusion criteria of the current code and ignores any other information even if it is relevant to a different code in the scheme.

Here we focus on two of the seven codes. First, Expected Result (ER), a broadening of Comparison to Theory in [10, 11] to include any mention of a result matching an expected outcome. Second, Consistent Results (CR), a combination of Comparison with Others and Repeatability from [10, 11]). ER is an uncommon code that was even less common in subset II, the only subset collected at Post (see Table II). CR is a more common code, though the outcome balance varies across the subsets (see Table II). Two coders established a very high degree of inter-rater reliability for each code within each data subset: Cohen's kappa ranged from 0.85 to 0.96 depending on the subset and code. A kappa value of 0.8-1.0 is considered to be near perfect agreement [12]. In this paper we are primarily concerned with developing the machine learning methods to perform automated coding so we will not discuss the educational significance of these two codes.

TABLE II: The proportion of responses that contained each code for each subset, according to human coders.

Subset	Expected Result	Consistent Results
I	0.35	0.82
II	0.07	0.53
III	0.23	0.66

C. Machine learning methods

To prepare the responses for automated coding, we used the following protocol on the full data set: i) split all words in each response into individual words (called tokens), ii) remove all punctuation, iii) remove any tokens that contain non-alphabetic characters (such as numbers), iv) remove stop words (such as "the" and "is"), and v) remove any remaining tokens that contain only one character [13]. A vocabulary was created out of all remaining tokens that occurred more than once, as exceedingly rare tokens have little predictive power. The vocabulary size was 1083 words.

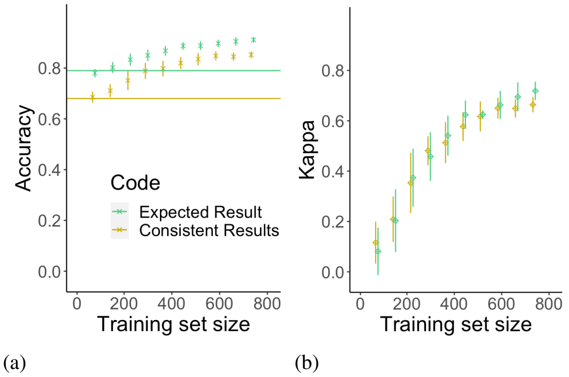
To convert student writing into numerical input for a neural network, an encoding method transforms words into a vector or matrix in a way that captures linguistic elements relevant in the coding scheme. We selected the Bag of Words binary encoding method. This is an NLP approach in which each response is transformed into a simple vector describing whether or not a word is present in the response. In preliminary tests using only subset I data, we found that Bag of Words binary encoding performed better than a few other encoding methods. Note that this and other coding schemes may be better modeled with a different encoding method.

In training, neural networks take in this numerical input and a numerical output and construct a model from the training data that maps inputs onto outputs. In this case the numerical outputs are 0/1 for absent/present, depending on whether the coder identified that the code was in the response. Separate neural networks were used to independently model the two codes. The neural networks were built in Python using a feedforward neural network model in Keras.

D. Evaluation of automated coding

Training data came from subsets I and/or II and test data always came from subset III. We evaluated the accuracy and reliability of the automated coding of subset III by comparing the machine learning algorithm's coding of 100 of the responses to human-generated codes. Each response in the data set could contain neither, one of, or both the ER and CR codes, therefore machine coding for each code was performed separately. For testing training set size, we randomly selected responses from the data subsets to create a training data set and tested performance across 5 neural network initial conditions. We repeated this analysis for 20 random training sets, and averaged across all 100 tests. For testing the data contexts, we selected three training sets of comparable size: subset I, subset II, and one randomly selected combination of the two. We averaged results for each across 10 neural network initial conditions.

We calculated accuracy and Cohen's kappa to evaluate our machine learning models. Accuracy, the fraction of correct code applications, is often the first metric reported by a machine learning algorithm performing a test. In content analysis coding, acceptable inter-rater accuracy is 0.80-0.90 de-



(a)

(b)

FIG. 1: (a) Accuracy and (b) kappa as a function of training set size (samples randomly pulled from subsets I and II combined). The horizontal lines in (a) illustrate the algorithms' first-order attempt at coding in cases of insufficient training. Error bars represent the standard deviations from repeated trials with randomized initial conditions and training data.

pending on the details of the coding scheme [14]. When researchers assess their ability to accurately apply a coding scheme to data, however, they often use Cohen's kappa instead to account for the random chance that two people would happen to code a response the same way. A kappa value of 0.6-0.8 is considered to be substantial agreement [12].

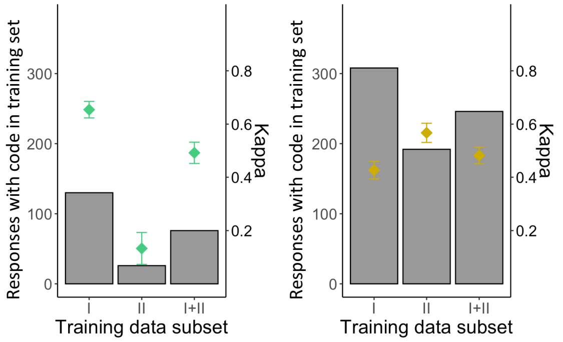
III. RESULTS

A. The effect of training size

Accuracy appears relatively high throughout the full range of training set sizes (Fig. 1). The accuracy for ER (a more rare code) is consistently higher than the accuracy of CR (a more common code), which we attribute to the different positive rates for each code.

When the training size was very low ($N = 73$), the machine coded very few responses as containing ER (on average, 1.5/100 true positives and 0.5/100 false positives). In contrast, it coded almost all responses as containing CR (4/100 true negatives and 1.5/100 false negatives on average). We interpret this to mean that, in cases of insufficient training, the algorithm's first-order attempt will be to code all responses with the most common outcome. Thus, a lower bound on accuracy is the condition where all responses are coded as containing CR and none are coded as containing ER, which translates to accuracies of 0.79 and 0.68, respectively (Fig. 1). The higher outcome imbalance (much fewer than 50% positive instances of the code) for ER means the accuracy for ER is higher than for CR (closer to 50% positive instances) for all training set sizes.

Compared to accuracy, Cohen's kappa is able to better account for outcome imbalances. This metric starts off very low for both codes at low training set sizes and grows quickly as the training data set size is increased, with significant overlap between codes (Fig. 1). We infer that Cohen's kappa may be a more descriptive and conservative metric for identifying conditions that allow for accurate predictions. Kappa starts to level off, while maintaining continuous increases, once consistent substantial agreement is achieved (N ≈ 600 for both codes, with 80-100 positive responses for each code).



(a) Expected Results

(b) Consistent Results

FIG. 2: The number of responses coded with (a) expected results and (b) consistent results in three different types of training set (bars) and the corresponding kappa (points).

B. The effect of training and test set context

Next we examine the reliability of tests from three types of training data: all subset I data (N=376), all subset II data (N=361), and a random sampling of responses from both I & II (N=361). We selected 361 responses at random from the full set of subsets I & II so that the training set size would be comparable across the three training data types. The models constructed by the three types of training data were applied to an identical set of 100 responses from subset III.

We see that Cohen's kappa depends on the type of training data (Fig. 2). For ER, using only subset I is most effective, followed by the random set selected from both I & II. Using subset II alone resulted in a low kappa value, likely because ER was rare in that training set. For ER, training sets perform better with more examples of the code.

This pattern of higher reliability with more positive examples, however, does not carry over to the automated coding of CR (Fig. 2). This code is much more common in all training set types compared to ER (Table II). In this case, the training data set that performs the best is subset II, where CR is least common. Moderate agreement can more readily be achieved because positive examples of this code are more common. We expect the algorithms over-predicted CR in subset I and the random I + II subset.

IV. DISCUSSION

Reliability increases with size. We find that, with sufficient data (training set size $\approx$ 600 responses with 80-100 positive responses for each code), neural networks are able to consistently achieve substantial agreement with human coders. Furthermore, the level of agreement continuously increases as more data are added to the training set. While accuracy as a function of training set size depends on the frequency of the code in the data set (outcome imbalance), reliability (via Cohen's kappa) as a function of training set size does not depend on this balance.

Results from previous studies are consistent with our findings of the relationship between training data size and kappa. Studies with very small data sets (N = 67 [6] and N $\approx$ 150 [7]) do not consistently reach moderate agreement, instead finding a wide range of agreement levels (0-0.7) across different codes. Consistent higher kappa values are achievable, however, when N is much higher or when unsupervised methods are used in the development of the coding scheme [4, 5].

Automated coding of data from a different context than the training data can be performed, but take caution. Most studies involving machine learning use portions of the same data set for both training and testing [4, 6, 7]. In this study, we were interested in developing methods capable of machine coding large-scale data sets across institutions and contexts. For example, we are interested in understanding how a machine learning algorithm would perform in coding data from an entirely new cohort of students or students under different experimental conditions. In this study, all tests were done using a different cohort of students in the training data set (either a mix of subsets I & II or one of these individually) as in the test data set (subset III). Figure 1 demonstrates it is not necessary to have data from the same population of students in the training and test sets to perform reliable automated coding. For both codes, sufficient reliability was achieved around N = 600 despite no subset III data in the training set.

Some characteristics of the training set, however, do impact reliability (Fig. 2). For ER, kappa was highest for the subset I training set. Kappa was lowest for Subset II because this training set contained very few examples of ER, likely because the instructional goals of the lab course are in line with decreasing the likelihood that students hold ideas in line with the ER code. There are two possible explanations for this pattern of reliability: first, the more balanced the presence of the code in the dataset (i.e., approximately 50% of responses contain the code), the better the reliability; second, that Pre-class data are better at predicting Pre-class data. To distinguish between these explanations, we turn to CR, a code that is present in the majority of responses.

For CR, kappa was much more consistent with moderate agreement across all training set types. This is likely because more examples of this code were present in all the training sets compared to ER. The highest performing training set, subset II, is the one that used only Post data, rejecting the idea that Pre data are inherently superior for predicting other Pre data. Another possible explanation for the high performance of subset II training data may be that subsets II & III were asked in a survey and thus both had shorter average response lengths, but this effect was not also seen for ER. Alternatively, the improved performance may be because subset II had the most balanced outcomes (present in approximately 50% of the responses) of all the training sets. Subset I, in contrast, had an overabundance of the code in the data set and had the lowest reliability. Though it may seem counter-intuitive that the training set with the most examples performed the worst, the overabundance of the response means that the algorithm is not as well trained on what is *excluded* from the code.

Thus our data show that **imbalanced outcomes in the training data are the primary factor in determining decreased reliability**. Different characteristics of the training set (e.g. Pre vs. Post) mostly matter in so far as they affect the balance of outcomes. More testing across other types of questions and codes is needed to generalize this conclusion.

V. CONCLUSION

We have examined the utility of a supervised machine learning algorithm for use in machine coding of responses to an open-ended survey question. Though the functionality of machine coding is highly dependent on the context of the question asked and the population of students in the data set, some takeaways from our particular analysis may be broadly applicable:

- You need more than about 80-100 responses that contain each code before you can consistently see substantial agreement with human coding. For coding schemes and data like ours, substantial agreement is consistently achieved around N = 600 total responses.
- You can successfully use training data from previous cohorts of students to code new data, but changes that alter the balance of outcomes (including those caused by different characteristics like Pre vs. Post) may decrease reliability.
- Optimal coding for binary schemes occurs when about half of the responses in the training data contain the code. Coding reliability decreases if the frequency of outcomes is too low or too high.

More research that uses different types of coding schemes for different types of student writing is needed to test the generalizability of these observations.

ACKNOWLEDGMENTS

This material is based on work supported by NSF Grant No. DUE-2000739. We thank Sophia Jeon for initial organization of the data. We would also like to thank Ruijie Jiang, Shuchin Aeron, and Brad Ramshaw for discussions related to this work.

[1] V. Otero and D. Harlow, Getting started in qualitative physics education research, in *Getting Started in PER*, Vol. 2 (2009).

[2] T. O. B. Odden, A. Marin, and M. D. Caballero, Thematic analysis of 18 years of physics education research conference proceedings using natural language processing, Phys. Rev. Phys. Educ. Res. **16**, 010142 (2020).

[3] B. Sherin, N. B. Kersting, and M. Berland, Learning analytics in support of qualitative analysis, in *Proceedings of International Conference of the Learning Sciences, ICLS, 1* (London, United Kingdom, 2018) pp. 464–471.

[4] J. M. Rosenberg and C. Krist, Combining machine learning and qualitative methods to elaborate students' ideas about the generality of their model-based explanations, Journal of Science Education and Technology **30**, 255 (2021).

[5] J. Wilson, B. Pollard, J. M. Aiken, M. D. Caballero, and H. Lewandowski, Classification of open-ended responses to a research-based assessment using natural language processing, Physical Review Physics Education Research **18**, 010141 (2022).

[6] M. Thomas, S. Bagley, and M. Urban-Lurain, Using machine learning algorithms to categorize free responses to calculus questions, in *Proceedings of the Twenty-first Annual Conference on Research in Undergraduate Mathematics Education* (2018).

[7] C. M. Nakamura, S. K. Murphy, M. G. Christel, S. M. Stevens, and D. A. Zollman, Automated analysis of short responses in an interactive synthetic tutoring system for introductory physics, Phys. Rev. Phys. Educ. Res. **12**, 010122 (2016).

[8] R. Jiang, J. Gouvea, D. Hammer, and S. Aeron, Automatic coding of students' writing via contrastive representation learning in the wasserstein space, Under review. (2020).

[9] N. T. Young and M. D. Caballero, Predictive and explanatory models might miss informative features in educational data, Journal of Educational Data Mining **13**, 31 (2021).

[10] D. Hu and B. M. Zwickl, Examining students' views about validity of experiments: From introductory to Ph.D. students, Physical Review Physics Education Research **14**, 010121 (2018).

[11] E. M. Smith, M. Stein, C. Walsh, and N. G. Holmes, Direct measurement of the impact of teaching experimentation in physics labs, Physical Review X **10**, 011029 (2020).

[12] J. R. Landis and G. G. Koch, The measurement of observer agreement for categorical data, Biometrics **33**, 159 (1977).

[13] J. Brownlee, *Deep Learning for Natural Language Processing*, v1.9 ed. (2021).

[14] J. W. Creswell and C. N. Poth, *Qualitative Inquiry and Research Design*, 4th ed. (SAGE, 2018).