Analysis of student essays in an introductory physics course using natural language processing

Amir Bralin
Department of Physics and Astronomy, Purdue University, 525 Northwestern Ave, West Lafayette, IN, 47907, U.S.A.

Jason W. Morphew
School of Engineering Education, Purdue University, 610 Purdue Mall, West Lafayette, IN, 47907, U.S.A.

Carina M. Rebello
Department of Physics, Toronto Metropolitan University, 350 Victoria Street, Toronto, ON M5B 2K3, Canada

N. Sanjay Rebello
Department of Physics and Astronomy / Department of Curriculum & Instruction, Purdue University, West Lafayette, IN, 47907, U.S.A.

We analyzed the essays that were written on various topics in an introductory physics course using two unsupervised machine learning algorithms. One of them was Latent Dirichlet Allocation (LDA). This algorithm is used for extracting abstract topics from a collection of text documents. The other algorithm was Non-negative Matrix Factorization (NMF). It is used for similar purposes but also in other domains such as image recognition. We applied these two algorithms to the dataset that consisted of $N = 683$ student essays. Although there were some built-in, important differences between LDA and NMF, they both found similar topics in our data by large. This offers instructors a promising and productive way of accessing useful information about their students' written work, especially in large-enrollment classes.

I. INTRODUCTION

This work contributes to the development of the approach to analyze student work using Natural Language Processing (NLP). The idea is to make machines read and process any text produced in a physics classroom and let humans engage in deeper analysis of that text. For example, students may write their solutions to a given physics problem. If the number of students in the class is too high, then the teacher will likely not have an adequate amount of time to evaluate them all in detail. To assist the teacher, one of the tasks which the NLP algorithms can do is text classification: each piece of text can be assigned a numerical value (e.g. 1 for being correct and 0 for being incorrect) or a categorical label (e.g. "kinematics", "statics", and "dynamics"). Other NLP-performed tasks include summarization, sentiment analysis, language translation, and more.

In this study, we focus on Topic Modeling – the NLP task to find themes or patterns within a body of text. Algorithms that can perform this task were developed during the 1990-2010s. A concise review can be found in Blei (2012) [1]. One such algorithm is called Latent Dirichlet Allocation (LDA). It was published by Blei et.al. (2003) [2] and has become popular since. LDA models human text **statistically**. Consider a text file or document as the collection of words. Each word is not random, but is associated with the topic of the document. The problem is that we don't know this topic a priori. Moreover, we have a collection of many documents written on various topics. That is why they are referred to as "latent" or "hidden". Instead, we know the posterior fact – the words produced by each topic. Using probability theory, this problem can be solved with Bayes' theorem. The name Dirichlet (1805-1859) is present in the name of this algorithm because it uses the Dirichlet distribution for inference and parameter estimation. LDA works best for large corpora of text documents. For example, 5577 articles published in the journal *Science Education* were analyzed with LDA in Odden et.al. (2021) [3]. A similar work but with a smaller dataset – 1302 papers from *PERC Proceedings* – was done earlier, by Odden et.al. (2020) [4]. A related work which had a comparable data size with ours is Geiger et.al. (2022) [5]. Its authors analyzed student responses to a conceptual question about the electric circuits in an introductory physics class. Using LDA, they obtained some key student ideas that were common to many of those responses. This opened a way to quickly generate useful insight into student thinking. In addition, there was a recent publication by Wilson et.al. (2022) [6]. The authors analyzed student responses to a conceptual survey using NLP techniques other than LDA. They achieved good agreement between their NLP-generated results for categorizing student responses and the ones obtained by traditional means.

To expand upon this line of research, we implemented LDA in our context and did the same for another NLP algorithm called Non-negative Matrix Factorization (NMF). It was developed in the 1980s under different names, but gained popularity and its present form in Lee & Seung (1999) [7]. Like LDA, NMF also models human text simply as the collection of individual words (also known as the "Bag-of-Words" approach in NLP). However, it arranges these (digital) words in a structured, tabular format – a matrix – and then uses the standard techniques of linear algebra. This makes NMF qualitatively different from LDA as the results (topics) are determined by the data (words) alone, without any statistical distributions involved. Then, our two research questions were: (1) what common topics can be found by both LDA and NMF in our data? (2) what are the differences between the results of these two algorithms?

A. Course context

This study took place at a land grant U.S. Midwestern university. All data came from over 2,000 undergraduate students enrolled in an introductory calculus-based physics course for engineers during the Fall 2022 and Spring 2023 semesters. The student population of the course included 25% women, 10% underrepresented minorities, and 8% international students. In terms of specialization, approximately 80% of the students were majoring in engineering.

The course was titled "Modern Mechanics". Based on the textbook "Matter and Interactions" (Vol.1) by Chabay and Sherwood [8], its content was centered around three fundamental concepts of classical mechanics: Momentum, Energy, and Angular Momentum. The course consisted of 3 main parts: lecture, recitation, and laboratory. Lectures were conducted in a traditional format: during two 50-minute-long sessions per week, in a large auditorium. Recitations served as the problem-solving sessions where the students worked in groups to solve problems similar to their exam questions. The **laboratory** setting was the source of data for our research. All students were divided into groups of three and each group worked as a unit. At the end of each 110-minute-long lab, one person from each group submitted the lab work on behalf of the group. No data thus collected included any personal, identifiable information.

B. Research context

During the last five weeks of the course, students were given a lab assignment: to find a real-world problem related to the course content and write an **essay** about how to solve it. The writing instructions were provided gradually over the span of the assignment: students started brainstorming about the problem during the first week, then came up with a solution during the next week, and iterated through this problem-solving cycle until the final week of the course.

Computation was an integral part of the labs. Our students were taught the basics of programming (such as variables and loops) using `Python`. During every lab session, the students also engaged in simple data analysis using the standard computational libraries `NumPy`, `pandas`, and `matplotlib`.

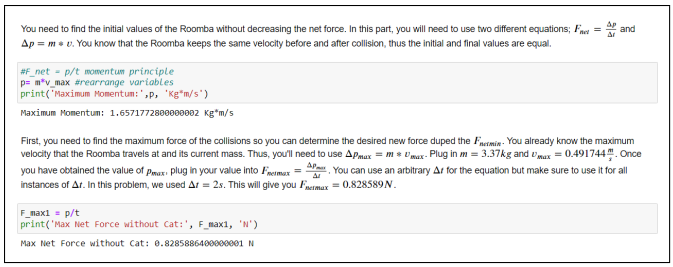

FIG. 1. A fragment from an example essay written by the students for their assignment. Note that there are short code cells alongside normal text with some equations.

As for the lab notebooks where they had to report their results, the Jupyter Notebook format was used. It is a web-based interactive writing platform which incorporates computer code alongside text. This allowed our students to create professional-looking lab notebooks that integrated both their written work and calculations seamlessly. A snippet from one student essay is shown on Figure 1 as an example.

II. METHODS

There were two algorithms that we applied to our data and compared: Latent Dirichlet Allocation (LDA) and Non-negative Matrix Factorization (NMF). Both are widely used for Topic Modeling: searching for common sets of words within a text by a computer. The usage of the term **set** above is important. In computer science, the key property of a set of items is that the order of those items doesn't matter. In contrast, a collection of items with some order defined for those items is called a sequence. The algorithms and data structures of Topic Modeling operate with sets, not with sequences. This means that when analyzing a human-generated text, both LDA and NMF ignore the order of words and sentences (the Bag-of-Words approach) in that text. This is a great assumption on the part of these models.

A. Topic Modeling

The workings of LDA were well-described in the aforementioned works of Odden and colleagues (Ref. [3], [4]). This algorithm represents the final product of a decade-long effort in the field of latent semantic analysis. The mathematical foundations of LDA are beyond the scope of this work. We only emphasize that it is a statistical model. As such, it needs a lot of data. Searching for an alternative that can remedy the shortage of data, we found NMF. In the original work [7], the authors Lee and Seung showed that it was able to extract certain semantic features from encyclopedia articles. The example they gave was the set of words {president, congress, power, united, constitution, amendment, government, law} which represents a clear, coherent topic.

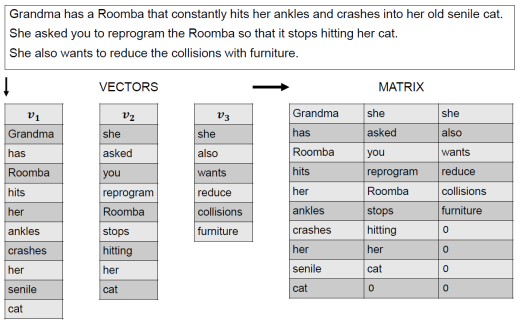

FIG. 2. An example of how vectorization works. Three sentences are given as input. After breaking each sentence into individual words (excluding articles and prepositions), it can be rewritten as a table column or a vector. Combining all three vectors corresponding to the three separate sentences results in the matrix corresponding to the original text. Here, all empty matrix elements are assigned number 0 while the original words remain the same for convenience. In fact, each unique word from the text is transformed into an integer number in the final matrix.

NMF belongs to the family of algorithms used for factor analysis. An algorithm from the same family that was widely used in PER and Science Education is Principal Components Analysis. In the context of NLP, factor analysis becomes relevant when the text data that we want to analyze takes the tabular, **matrix** form. This can be done by constructing the matrix $V = \{v_1, v_2, ..., v_n\}$, where v_i is the text of the i-th document ($i = 1, ..., n$) in the vector form. This is called "vectorization" or tokenization and is shown on Figure 2 as an example. Here, n is the total number of documents in the dataset. Then, NMF finds the simplest possible model of V: the product of two other matrices W and H, by performing matrix decomposition and minimizing the difference (Euclidean distance or Frobenius norm) between V and WH. The matrix V is called "visible" because it contains the actual, observable data. The matrix H is called "hidden" because it contains the values which we cannot observe directly. The matrix W contains the "weights" of the visible variables with respect to the hidden ones.

B. Data Processing

In total, $N = 683$ student essays from two semesters were collected. All essays were retrieved from the university's online learning platform. No limitation on the size of the essay was explicitly stated. The distribution of the essay lengths measured by the total number of words is shown in Figure 3. On average, an essay was approximately 1,000 words long.

Our data cleaning included the following: (i) discarded all lines of code written by the students in order to only work with text; (ii) discarded all numbers and mathematical symbols in order to keep only the text in English; (iii) parsed the remaining raw text into individual words;

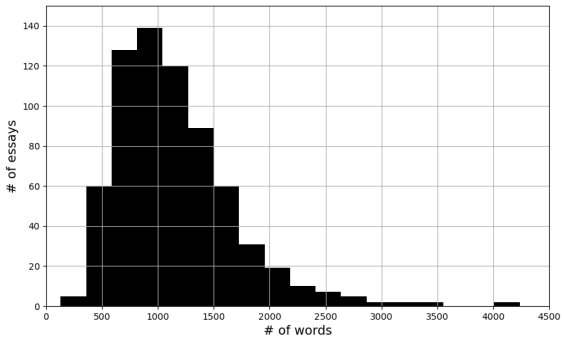

FIG. 3. The histogram of essay lengths measured by the word count. The total number of essays was $N = 683$. The number of bins used for plotting was 30. The mean and standard deviation were $\mu = 1,170$ and $\sigma = 595$, respectively.

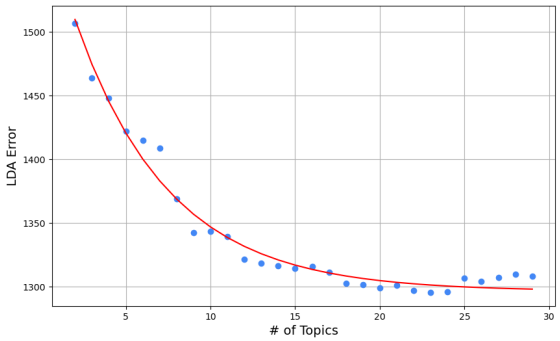

FIG. 4. The error rate vs the number of topics used for topic modeling with LDA. The function (red line) used for fitting the data points (blue dots) on the plot: $p(n) = 1,296 + 305.7e^{-0.18n}$. The graph decreases until "saturating" at $p \approx 1,300$.

(iv) disposed of articles, prepositions, and pronouns among those words in order to keep nouns, adjectives, and verbs. Our data analysis was done in `Python`. Specifically, we used the library scikit-learn [9] for Topic Modeling. Both LDA and NMF algorithms were accessed through its `sklearn.decomposition` module. The intermediate step between data cleaning and analysis was vectorization (see Section II A): converting the essay words into digits that a model (LDA or NMF) can operate with. This was done using scikit-learn's `CountVectorizer` (for LDA) and `TfidfVectorizer` (for NMF) sub-modules. The Python code that served as the basis for our analysis was taken from scikit-learn and can be found at Topic Extraction with NMF and LDA.

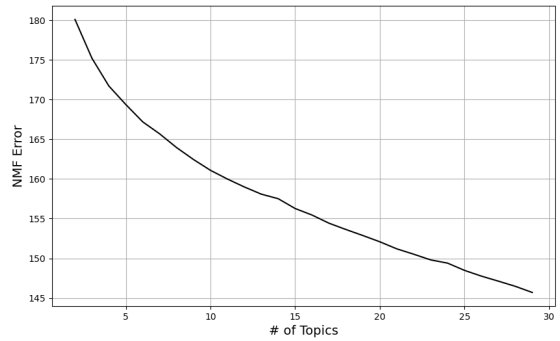

FIG. 5. The error rate vs the number of topics used for topic modeling with NMF. The graph decreases monotonically.

III. RESULTS

First, we modeled our data with LDA. We assigned a fixed number of topics to the LDA algorithm and it found them from the essay dataset. To evaluate its precision, we plotted the associated error vs the number of topics on Figure 4. As depicted, LDA reached a limit at n = 20-30 after which, increasing this number did not result in any significant improvement. Then, we modeled our data with NMF. The corresponding plot of the associated error is shown on Figure 5. It didn't reach any limit by increasing the number of topics assigned to the model. Rather, it decreased monotonically. This is a built-in property of NMF. In principle, its error would reach 0 if the number of topics was equal to the number of essays in the dataset. Such perfect precision, however, wouldn't help with accuracy.

The $n = 10$ topics found by LDA are given in Table I. They are represented by the five most likely words associated with each topic. For example, Topic 9 reads {"wheel", "friction", "mass", "braking", and "power" }. From this set of words we can understand or infer a that this topic is about cars. Possibly, about ensuring the safety of a car's braking mechanism by studying the friction between its wheels and the ground. In principle, there are as many words associated with each topic as there are words in the entire dataset. However, a meaningful LDA topic may be represented by a few words which have the highest probability of occurring in an essay. By glancing at other columns in the table, we can see that not all of the topics are as clear. For example, under Topic 7, there is a word "shirt" that doesn't fit the context of that topic. In addition, there are several words (such as "ball", "earth", and "car") that appear in multiple columns. This further complicates the **interpretability** of LDA.

The $n = 10$ topics found by NMF are given in Table II. Although some words (e.g. "momentum") are still found in more than one column, they are all general terms expected to be present in multiple topics. More specific terms such as "ball", "earth", and "car", which have clearer associations in the human mind, are now localized: each one appears in a single column.

TABLE I. $n = 10$ topics modelled by the Latent Dirichlet Allocation (LDA) when applied to the essay data. Each column represents a topic, defined by the 5 "top words" that are most associated with it.

Topic 1	**Topic 2**	**Topic 3**	**Topic 4**	**Topic 5**	**Topic 6**	**Topic 7**	**Topic 8**	**Topic 9**	**Topic 10**
car	bungee	spring	car	ball	drone	pool	rocket	wheel	rollercoaster
speed	mass	constant	collision	angle	time	cue	earth	friction	loop
distance	change	change	change	distance	mass	collision	mass	mass	cart
change	speed	mass	ball	height	drag	ball	change	braking	height
wall	time	height	mass	air	air	shirt	train	power	friction

TABLE II. $n = 10$ topics modelled by the Non-negative Matrix Factorization (NMF) when applied to the essay data. Note that the order of NMF topics is not the same as that of LDA nor the topics themselves are identical in two tables.

Topic 1	**Topic 2**	**Topic 3**	**Topic 4**	**Topic 5**	**Topic 6**	**Topic 7**	**Topic 8**	**Topic 9**	**Topic 10**
car	rollercoaster	ball	energy	rocket	spring	train	drone	braking	wheel
collision	loop	angle	force	asteroid	bungee	stop	parachute	force	turbine
force	cart	player	device	satellite	bullet	energy	terminal	rider	energy
energy	ride	velocity	angular	fuel	toy	force	delivery	stop	water
change	height	hit	torque	earth	energy	hill	force	brake	wind

IV. CONCLUSION

This study aimed at contributing to the methodology in PER that is based on using the computational resources offered by Natural Language Processing (NLP). To this end, we performed Topic Modeling on student essays in an undergraduate physics course about introductory mechanics. We used two different algorithms for the same dataset: Latent Dirichlet Allocation (LDA) and Non-negative Matrix Factorization (NMF). The former is a statistical model of digital text treating each topic within that text as a distribution of text's words. The latter is a model of digital text which treats it as a matrix of text's words and performs factor analysis on it. Both models offer a quick and powerful way of interpreting class materials: not just essay documents, but any other text files that need to be read and processed. These can be homework assignments, quiz responses, or the answers to some open-ended exam questions. In our context, a teacher may see common themes among the essays prior to reading them. This may save some time and effort in getting familiar with the student work and grading it later.

We found that for our dataset, in particular, NMF offered a complimentary way of interpreting the topics found by LDA. This method of Topic Modeling is the domain of unsupervised machine learning which is hard to evaluate. We expect that one could combine the two algorithms into a single, unifying procedure: fit the entire dataset on LDA, then fit a subset of the data on NMF to cross-validate the LDA results. Overall, the common themes among the topics found by both LDA and NMF are in agreement with each other. The difference in the interpretability of the topics between the two models becomes evident upon a more detailed evaluation of each topic.

V. LIMITATIONS AND FUTURE WORK

A careful reader will notice that the absolute values of the error rate for LDA and NMF in Figures 4 and 5 are not comparable. This means that we cannot compare the performance of these two algorithms directly, by observing the error decreasing with increasing number of topics. Thus, we tried to compare the topics found by LDA and NMF manually, by looking at the corresponding Tables I and II.

In using Topic Modeling for our student essays, we completely disregarded the code and math symbols that were present in the original data. Neither LDA nor NMF are suitable for analyzing these aspects of language in their default modes. Future work in this direction will require either adjusting these two algorithms accordingly or using other NLP methods such as large-language models.

There is a fundamental discrepancy between NLP-based and human interpretation of written text. The topics generated by a model of choice (whether LDA, NMF, or any other) are not related to the topics that a human being may find in the same body of text. The former are mathematically defined constructs that represent our best estimate of what constitutes a "topic". The latter are certain concepts in our minds which may or may not coincide with these estimates. Therefore, we do not suggest that this methodology can completely replace the traditional human analysis of educational texts.

VI. ACKNOWLEDGEMENTS

This work was supported in part by U.S. National Science Foundation Grant 2021389.

[1] D. M. Blei, “Probabilistic Topic Models: Surveying a suite of algorithms that offer a solution to managing large document archives.”, Commun. ACM 4, 55 (2012), pp. 77–84.
[2] D. M. Blei, A. Y. Ng, M. I. Jordan, Journal of Machine Learning Research 3 (2003) pp. 993-1022.
[3] T.O. B. Odden, A. Marin, and J. L. Rudolph, Science Education 4, 105 (2021), pp. 653-680.
[4] T.O. B. Odden, A. Marin, and M. D. Caballero, Phys. Rev. Phys. Educ. Res. 1 , 16 010142 (2020).
[5] J. M. Geiger, L. M. Goodhew, and T.O. B. Odden, *Physics Education Research Conference 2022*, pp. 206-211.
[6] J. Wilson, B. Pollard, J. M. Aiken, M. D. Caballero, and H. J. Lewandowski, Phys. Rev. Phys. Educ. Res. 18.010141.
[7] D. D. Lee and H. S. Seung, Nature 401 (1999), pp. 788-791.
[8] R. Chabay, B. Sherwood, Matter and Interactions (2015), John Wiley & Sons, 4th edition.
[9] F. Pedregosa et.al. *Scikit-learn: Machine Learning in Python*, Journal of Machine Learning Research 12, 85 (2011).