Improvement of Hybrid Feature Identification and Machine Learning Algorithms in Automatic Essay Scoring systems

Weifeng Jin
University of Washington
LING 575 Spring 2018 Final Report

***Abstract*— This project report discusses several possible ways of hybrid feature identification and machine learning algorithms to improve the performance of current Automatic Essay Scoring (AES) systems. The report would also briefly discuss the history of different AES systems and introduce the basic mechanics of them. Most of the suggested enhancements are based on the Electronic Essay Rater (E-Rater), developed by Educational Testing Service and deployed in GMAT AWA task. These improvements aimed to increase the features used in scoring stage and can be added into three different modules, syntactic, rhetorical/discourse and topical.**

I. INTRODUCTION

The automatic essay scoring task is the use of specialized computer programs to assign grades to essays written in an educational setting, including public exams and coursework. On most occasions, the grading scale in such settings is a small number of discrete categories. For example, scores for the GMAT AWA task range from 0 to 6 in half-point intervals. Hence, the task itself is a variation of statistical classification, which aims to classify a large set of texts into these specific score categories. There are several significant factors contributing to the development of AES system, including accountability, standards, technology and cost. That is to say, the target of developing an AES system is to achieve a high accountability for both test administrators and test-takers, establish a rigorous and accurate standard, utilize the current technology and lower the cost as much as possible.

II. HISTORY OF AES

A. Project Essay Grader (PEG)

The first AES system in the world of computational linguistics, the Project Essay Grader (PEF) was developed by Ellis Page in 1968. PEG uses proxy measures to grade student-written essays, including average word length, essay length, number of semicolons and commas. Like most current AES systems, PEG has two stages in grading, training and scoring. The drawback of the system is that it ignores "the semantic aspect of essays and focusing more on the surface structure [1]" and it is not cost effective.

B. Intelligent Essay Assessor (IEA)

Intelligent Essay Assessor (IEA) was developed by Peter Foltz and Thomas Landauer in 1997 and is now a product of Pearson Educational Technologies. In addition to measuring grammar correctness and punctuation, the system also tries to employ systematic semantic text analysis using Latent Semantic Analysis (LSA).

C. Electronic Essay Rater (E-Rater)

Electronic Essay Rater (E-Rater) was first used commercially for scoring of GMAT AWA task in February 1999. The system was developed by Jill Burstein in Educational Testing Service. It uses a corpus-based approach to model-building in both training and scoring stage, in which actual essay data are used to train the model [2]. Since then, the score for the AWA task is determined through e-rater and one human-scorer, and the discrepancy between them has been less than 3%. E-Rater divides the feature identification stage into three different modules: syntactic, rhetorical/discourse and topical. These three modules together would generate a special feature vector for the essay and various machine learning algorithms would be deployed to generate the statistical classification model. I would mostly base my improvement on these three modules in the feature identification stage.

III. TWO DATASETS

A. The Hewlett Foundation: Automated Essay Scoring

This dataset is used in a public competition in Kaggle.com held by Hewitt-Packard [3]. There are 8 essay sets and each of the sets of essays was generated from a single prompt. The length of essays range from an average length of 150 to 550 words. All response were hand graded and double-scored by real human graders. The reason that I use this dataset for the AES testing is that most testing services, including ETS, would not make their own essay corpus available to the public and this dataset has been proved to be a great representation of the AES task.

B. Spoken and Written English Corpus of Chinese Learners

This dataset is used by several scholars, including Xiaofei Lu in the Pennsylvania State University, to measure the syntactic complexity of a certain essay [4]. This corpus comprise 3554 essays written by English major students from 9 different universities in China [5]. These essays contain an average of 315 words. Since it has a score of syntactical complexity, it can be used to improve the syntactic module of the AES system. This dataset is not free for the public.

IV. Syntactic Module

The original syntactic module in the E-Rater system developed by ETS involves following steps [2]:

- A part-of-speech tagger is used to assign part-of-speech labels to all words in an essay.
- A syntactic "chunker" finds phrases and assembles phrases into trees based on sub categorization information for verbs.
- Identify various clauses, including infinitive, complement, relative, and subordinate clauses and occurrences of modal verbs and count the numbers.
- Use the number of these syntactic features to calculate ratios of them per sentence and per essay.

A. Limits of the System

As far as I am concerned, there are two major limits of the syntactic module of E-Rater. First and foremost, there are not enough syntactic features to make a reliable judgment for the syntactic complexity of the essay. E-Rater only uses the occurrence of certain syntactic structures to represent the grammar correctness, but most natural language processing systems would use a more complicated feature vector to perform similar tasks. In this case, the E-Rater syntactic module does not consider the length of sentences, clause embeddedness the use of complex nominals in the essay.

Another limit is that the system, at least in its current description, is that it does not consider cases of misspelling and wrong grammar use. If a misspelled word (an Out-of-vocabulary word) or a ungrammatical sentence is passed into the parser, it is highly possible that there would be no viable parsing trees at all. If that happens, the syntactic module would not be able to gather the required syntactic information from the structure and the sentence would be threw away. Some degree of tolerance should be given to such cases in order to fairly judge the performance of the student in an essay.

B. Improving the Syntactic Module

As shown by Lu [4] and the PEG model, some proxy measures can be used to measure grammar correctness, including mean length of clauses, mean length of sentences, dependent clauses per clause, and complex nominals per clause. These features can be added into the vector to measure the syntactic variability. In addition to that, T-unit, a concept coined by Kellogg Hunt in 1995, defined as the shortest grammatically allowable sentence, can be useful in measuring the syntactic complexity. Thus, features as mean length of T-unit clauses per sentence, clauses per T-unit, complex T-units per T-unit, dependent clauses per T-unit, complex nominals per T-unit and verb phrases per T-unit can be included in the vector.

Lu also pointed out these features have a high correlation with the syntactic complexity of certain essay shown in the SWECCL corpus respectively [4]. These preliminary findings show that these features are potentially great candidates for the feature vector representation.

Besides, it is necessary to improve the tolerance of the parser with certain penalties on the writer. For example, if there is a word that is an OOV, the system can tag it and penalize the author for the misspelling. If the part-of-speech tags cannot be parsed into a tree, the system can try to match a similar tree and penalize for the difference. Also, same/similar errors should only be penalized for once, which makes the system more lenient and user-friendly. The error-tolerance standard can be customized according to the test requirements.

C. Methods for Improvement Verification

Lu has shown that the additional features all have high correlation with the syntactic complexity [4]. However, it remains unsolved that if these features together would be a good indicator for grammatical variability. It is necessary to test the data in the SWECCL corpus since it has been scored by syntactic complexity. Since the dataset is only available in CD-ROM in certain markets, I have not been able to get the dataset and tried the features.

The error-tolerance system may not have significant effect in improving real performance of the AES system. When grading by human graders, small errors might just be ignored and big errors would always accompany other serious issues leading to a low score. It is inevitable that test-takers would make some mistakes, and the neural system here is more fair than the original one. Also, the ability of customizing the system a little bit adds some useful functionality to the AES system.

V. Discourse Module

The discourse module, also known as the rhetorical structure analysis, partitions the passage into several argument blocks and passes this information to the topical analysis module to generate feature vectors. The original discourse module in E-Rater involves following steps:

- The system uses rhetorical cue words and structure features, in addition to other topical and syntactic information, to identify discourses throughout the passage.
- The Automated Argument Partitioning and Annotation (APA) system partitions the passage into several different arguments. For every argument, the argument unit is labeled as "marking the beginning of an argument" or "marking argument development. [6]"
- These argument-partitioned version of essays are passed to the topical analysis module to evaluate the content of individual arguments respectively.

A. Limits of the System

The system relies heavily on cue words to establish the initial discourse analysis of the essay, which can be both inaccurate and misleading on some occasions. Additionally, the original APA system only classifies the argument units into 2 categories: beginning and development, which are clearly not enough for making a convincing argument.

Besides, the APA partitions the argument according to paragraph breaks rather than the logic flow. Thus, it might

make a wrong judgment when the argument spans over two paragraphs.

B. Improving the Discourse Module

In order to address the limits of the original discourse module, we have to find better ways to partition the essay into separate argument units. Lawrence suggested that discourse indicators (including cue words used in E-Rater) can be used to classify each clause as major claim/claim/premise/non-argumentative [7]. With more specific argument categories, the partitioning system would generate a better representation of the author's intended argument structure. Examples of discourse indicators for two relation types, Support and Conflict are shown below:

Relation Type	Words
Support	because, therefore, hence, for, since, assuming, so, thus, consequently
Conflict	however, but, though, although, except, never, nonetheless, yet, despite

In addition to that, we can use argumentation scheme structure to improve the partitioning results. Douglas Walton's book Argumentation Schemes for Presumptive Reasoning has established a great number of argumentation schemes [8]. Two modified example argumentation schemes for partitioning is given below:

Argument From Popularity
Premise: A large majority accept proposition A as true.
Premise: If a large majority accept A as true,
then there exists presumptions in favor of A.
Conclusion: There exists a presumption in favor of A.
Argument From Cause to Effect
Premise: If A occurs, then B will occur.
Premise: A occurs.
Conclusion: B occurs.

As you can see, there is one or several possible argument units in a single argument structure. If we can fit the argument units in the essay into the scheme, the structure of the essay would be clear and well-informed. The difficulty of this part is to determine the general argument type of each specific structure in the essay, which could be inaccurate on some occasions.

C. Verification of Improvement

Since the discourse module does not generate specific numbers for performance evaluation, the evaluation of improvement would be qualitative, namely whether the system can accurately divide the essay into right argument partitions and mark each argument unit correspondingly.

The annotation would be troublesome and actually has variations between different annotators as people have different standards on what qualifies as an argument. I attempted to mark the argument partitions on a subset of the dataset to prepare for the future test and evaluation using my best judgment. I classify each clause into major(major claim)/claim/premise/non(non-argumentative). Most essays have 2-3 different arguments supporting their main ideas. It is interesting to note that for essays with higher scores, it is much easier to navigate through the essay and search the respective arguments. and evaluation. The annotations can be found in: `github.com/wfjin/LING-575-AES-Improvement/edit/master/dataset-argument-units`, and an example of an annotated example (partially) is given below:

[major]Dear local newspaper, I think effects computers have on people are great learning skills/affects**[/major] [premise]**because they give us time to chat with friends/new people**[/premise]**, **[premise]**helps us learn about the globe(astronomy)**[/premise]** and **[premise]**keeps us out of troble!**[/premise] [non]**Thing about!**[/non][non]** Dont you think so?**[/non] [premise]**How would you feel if your teenager is always on the phone with friends!**[/premise][premise]** Do you ever time to chat with your friends or buisness partner about things.**[/premise] [claim]**Well now - there's a new way to chat the computer, theirs plenty of sites on the internet to do so: @ORGANIZATION1, @ORGANIZATION2, @CAPS1, facebook, myspace ect. **[/claim]**

VI. Topical Analysis Module

The topical analysis module evaluates the lexical and topical content of an essay by comparing the words it contains to the words found in manually graded training examples. Hence, it is a task of calculating word similarity between the target essay and the training dataset. There are two major programs in this module: EssayContent, which calculates the essay similarity based on word frequency, and ArgContent, which calculates the essay similarity based on word weight [2].

- **EssayContent**:
  - The program converts the vocabulary of each score category to a single vector. The elements of the feature vector represent the frequency of each word in the corresponding essay set.
  - The program converts the vocabulary of the target essay into a single vector in the same way.
  - The program computes the cosine similarity between the test essay vector and vector for each score category and assigns the score with the highest similarity to the essay.
- **ArgContent**:
  - The program assigns scores to each argument in the test essay generated by the Discourse Module.
  - The word frequency vectors are converted to vectors of word weights. The weight for word i in score category s is: $\mathrm{w}_{i,s} = (\mathrm{freq}_{i,s}/\mathrm{max\_freq}_s) * \log(\mathrm{n\_essays}_{total}/\mathrm{n\_essays}_i)$
  - The program evaluates each argument using cosine similarity and uses the adjusted mean of the set of argument scores for the score of the test essay.

Scores calculated from above two programs would serve as the preliminary score for the topical content of the test essay. These scores would be passed into the final machine learning algorithm to determine the overall essay performance.

A. Limits of the System

There are several drawbacks in the topical analysis module in the E-Rater system. Firstly, although the argument unit partitioning is passed along from the discourse analysis module, the topical analysis module does not check the logic flow and reasoning of the argument and assign score to the logical thinking of the essay, which is also part of the essay scoring process. In addition to that, when calculating the weights for the scores, the module does not take the repeated usage of certain words into consideration.

The system relies heavily on the training corpus, since the score it assigns to the target assay is based on the similarity score between that and the essays in the corpus. Thus, for essay with new prompts, the module would not be able to calculate the similarity and the size of the corpus is another significant contributing to the accuracy of scoring.

B. Improving the Topical Analysis Module

It is extremely difficult to propose improvements over the topical analysis module since it mainly relies on the word similarity between the target essay and the corpus. Some trivial (possible) improving measures include changing the similarity type (for example, from cosine to euclidean similarity) and adjusting the word weights to achieve a better performance.

In order to measure the validity of logic reasoning of the essay, we have to use the argumentation partitioning information for the discourse module. As I have mentioned in the discourse module, we can use the Argumentation Scheme structure to measure the logic soundness. We can use the Argumentation Scheme Structure to score the score of the logical part, including the structure completeness, easiness to navigate and scheme variability. These features can be used as indicators for logic reasoning scoring. When using the ArgContent and word weights information to calculate the score for each argument, we should also consider potential use of

C. Verification of the Improvement

These improvement measures proposed above also requires a huge corpus for training and testing. Thus, it can also be extremely difficult to perform the verification of the improvement due to the lack of adequate corpus and training algorithms. Both the dataset provided by HP and the SWECCL corpus can be used to calculate the word similarity. Using the improving discourse analysis module, we can also generate a set of partitioning information attached to each essay and for each argument, there would be a proper argumentation scheme attached to it. These partitioning information and argumentation schemes can be used to score the logical reasoning of the essay.

The verification would be left as something to do in the future work and would be discussed further in the relevant section.

VII. Training and Testing Using Machine Learning Algorithms

After the feature identification process, these features generated for a single essay would be quantized and incorporated into a single feature vector to represent the target essay. The original E-Rater system, using the syntactic, discourse and topical analysis module, would generate a total of 57 features for a single essay [2]. If implementing improvement measures mentioned in this report, there would be far more features presented for the training and testing stage.

The original E-Rater system uses a stepwise linear regression model to compute the optimal weights for theses features during the training stage and uses the weight model to compute the final score of the test essay. The weight model would be specific to each prompt and the selection criteria would vary a lot among different topics. ETS does not reveal what weights that algorithm might generate and the details of the implementation so that we do not know much about the training and testing stage in that sense.

Hence, it is hard to improve the training and testing without sufficient knowledge of the original system. It is possible to test different machine learning algorithms for prediction if we have the feature vectors. For example, on some occasions, support vector machines can have a better performance than the linear regression model. Some scholars also pointed out that a hybrid machine learning model, involving both linear regression and other discrete classification models, can have a satisfying performance in AES task.

VIII. Conclusions

The AES task is a complicated task involving knowledge of natural language processing, linguistics, mathematics, machine learning and computer science. NLP technology plays a significant role in the process of feature identification. The improvement of the syntactic module involves adding more features using proxy measures, and a modified parser with error tolerance and penalty to the grammatical analysis part. These features have a high correlation with the syntactic complexity and can be used as a good indicator of syntactic score in the feature vector.

Besides, the discourse module can be improved by using Argumentation Structure Schemes to enhance the argument partitioning and more discourse words can be used in addition to cue words. I also annotated a small subset of the dataset to be used as a training set for the argument partitioning task. After finishing the discourse module, the argumentation partitioning information would be passed to the topical analysis module. The topical analysis module can be improved by using the additional argumentation scheme information from the discourse module. Each argument would be scored and the logic reasoning would be inspected.

I believe these improvement measures would generally

benefit the E-Rater system and the accuracy of scoring would be significant improved.

IX. Future Work

It left a lot to be done in the AES improvement in the future. Firstly, these two datasets are still not sufficient for the improvement verification. In the future, the first task I should continue to annotate the dataset with argument unit classifications to make it more sufficient for the discourse module training task. After that, the annotated dataset can be used to improve the discourse module's partitioning performance.

Besides, the added feature for the syntactic module have not been verified for its performance. The SWECCL corpus has scores for syntactic complexity and can be used to measure the performance of these features. However, I have not been able to get the full dataset and in the future it should be in the checklist.

The topical analysis module improvement is also not verified and it is based on the new discourse module so that they can be incorporated in the future in terms of improvement verification. After the feature identification process, the machine learning algorithm should be used to assign optimal weights for each feature. In the future, I should try to use different algorithms and combine them if possible to find the best model in the task.

References

[1] S. Dikli, Automated Essay Scoring, Turkish Online Journal of Distance Education, vol. 7, no. 1, Jan. 2006.

[2] M. D. Shermis & J. C. Burstein (Eds.), Automated essay scoring: A cross disciplinary approach. Mahwah, NJ: Lawrence Erlbaum Associates., 2003.

[3] The Hewlett Foundation: Automated Essay Scoring — Kaggle. [Online]. Available: https://www.kaggle.com/c/asap-aes. [Accessed: 10-Mar-2018].

[4] X. Lu, Automatic analysis of syntactic complexity in second language writing, International Journal of Corpus Linguistics, vol. 15, no. 4, pp. 474496, 2010.

[5] Wen, Q., Wang, L. & Liang, M. 2005. Spoken and Written English Corpus of Chinese Learners. Beijing: Foreign Language Teaching and Research Press.

[6] J. Burstein, K. Kukich, S. Wolff, C. Lu, M. Chodorow, L. Braden-Harder, and M. D. Harris, Automated scoring using a hybrid feature identification technique, Proceedings of the 17th international conference on Computational linguistics -, 1998.

[7] J. Lawrence and C. Reed, Combining Argument Mining Techniques, Proceedings of the 2nd Workshop on Argumentation Mining, 2015.

[8] D. Walton, Argumentation Schemes for Presumptive Reasoning, Jan. 1995.