Automated Essay Scoring System

Zhuoyue Wang, zhuoyue2
Zhaoyu Wu, zhaoyuw2
Siqi Xiong, siqix2

Introduction:

In terms of current research, Automated Essay Scoring is perhaps one of the areas which have been over-looked. There hasn't been much of active research going on concerning essay scoring systems. While the critics argue that AES cannot possibly grade essays in a meaningful manner, however, they often fail to realize that while we, humans, can identify good pieces of writing, we often find it difficult to quantify or even articulate why a piece of writing holds value.

Automated Essay Scoring systems fill this gap by establishing a common, quantifiable basis for grading essays. The development of automated essay scoring systems is a common ground for test developers like ETS, cognitive scientists, and computer scientists. Previously, giants like ETS have worked extensively in AES and related areas, having the benefit of the huge datasets and models which get trained over every single applicant that registers for the test.

Inspired by the Kaggle competition "The Hewlett Foundation: Automated Essay Scoring", Wewant to develop an automated scoring algorithm and implement a basic machine learning model for graded student-written essays.

Problem definition:

Automated essay scoring system is the use of specialized computer programs to assign grades to essays written in an educational setting. It is a method of educational assessment and an application of natural language processing. Its objective is to classify a large set of textual entities into a small number of discrete categories, corresponding to the possible grades—for example, the numbers 1 to 6. Therefore, it can be considered a problem of statistical classification based on different linguistics features.

Our idea is to build an unsupervised essay scoring system that runs on extensive set of highly engineered and hand-crafted features to find out specific cues to grade essays in the cases where sample size could be as low as 10 graded essays. Then Wecan use some statistical learning techniques to find a best model to fit the dataset and predict essay scores.

Previous work:

1. Burstein, J., and Marcu, D. 2000. Benefits of modularity in an automated essay scoring system. ISI, University of Southern California (Association for Computational Linguistics).

This paper gives me a glance about the real world application of automated essay scoring (E-Rater in ETS) and provides an excellent description of how can automated essay scoring be implemented as evaluation supplement in real exams, the latest advances in the field and how

reliable they are. The E-Rater system is based on holistic scoring guide criteria. Holistic scoring guides instruct the human reader to assign an essay score based on the quality of writing characteristics in an essay. For instance, the reader is to assess the overall quality of the writer's use of syntactic variety, the organization of ideas, and appropriate vocabulary use. E-rater combines several NLP tools to identify syntactic, discourse, and vocabulary-based features. This module design inspires me to build a system with similar structure.

2. Attali, Y. and Burstein. J. 2006. Automated Essay Scoring With e-rater version 2. A publication of the Technology and Assessment Study Collaborative Caroline A. & Peter S. Lynch School of Education, Boston College.

This paper provides me more detailed explanation and implementation on the structure design on E-Rater system. The feature set used with E-rater V.2 include measures of grammar, usage, mechanics, style, organization, development, lexical complexity, and prompt-specific vocabulary usage.

This paper gives me more information about the design of Our latter modules. In the part of Organization and Development, The E-rater system uses the discourse elements in variety of ways, depending upon the type of prompt and the discourse strategy. In the case of persuasive or informative essays (Like Our dataset topic). It follows a discourse strategy that requires at least a thesis statement, several main and supporting ideas, and a conclusion. This inspires me to design Our Discourse Analysis Module.

Different from Our implementation, the E-rater cares about organization. It assumes a writing strategy that includes an introductory paragraph, at least a three-paragraph body with each paragraph in the body consisting of a pair of main point and supporting idea elements, and a concluding paragraph. So it will measure the difference between this minimum five-paragraph essay and the actual discourse elements found in the essay.

3. Salvatore, V., 2003. An Overview of Current Research on Automated Essay Grading. Universita' Politecnica delle Marche, Ancona, Italy

This paper presents an overview of current approaches to the automated assessment, which helps me find some great resource to implement in Our own system, including Electronic Essay Rater (E-Rater) and Project Essay Grade (PEG). Wehave described the features in E-Rater in the last point. For Project Essay Grade, it brings me to think about building a system that primarily relies on style analysis of surface linguistic features with a block of text. Thus, an essay is predominantly graded on the basis of writing quality, taking no account of content. In this way of thinking, it is more statistical rather than linguistic, because the system is more relying on the assumption that the quality of essays is reflected by the measurable proxies. Westarted with this statistical approach and finally it becomes Our first module.

Approach Description:

Weimplement an unsupervised essay scoring system, which works on a set of natural-language analysis modules which capture various salient features of essays. This system only requires 12 "best-quality" essays from the original dataset. Wewas initially looking into methods proposed in (Forman and Cohen 2004), which allow to built efficient systems learning from little data. But as

it turns out, even these approaches require a considerable amount of data which defeats its very purpose of having an unsupervised system. The system works out by scoring the essay on different trait-specific features which include syntactic, lexical and semantic cohesive devices. By tuning several statistical learning models, Wehave built a linear regression based model to help evaluate the final scores by fitting scores in five modules.

Implementation & Evaluation:

This model has been implemented in Python 2.7.13, and require machine-learning based libraries. The implementation is done in python and some related libraries, including NLTK, SciPy, NumPy and sklearn.

Weevaluate Our scoring system models by using classification accuracy and the root-mean-square error (RMSE), which can directly show the system performance.

Summary on the automated essay scoring system

The system runs on a modular architecture. Inspired by a modular design in (BURSTEIN and MARCU 2000), there are 5 modules which capture diverse cohesive devices of the text and quantify it based on hardcoded metrics. The test program aggregates all the output scores from the five modules and transmit them into a score civ file, which can be trained over a regression model by scoring model r file. The scoring model r file takes in all the scores from the modules and outputs a final score weighted appropriately.

Statistical Analysis Module

The module extracts statistical features from the essay including word count, longword count (words with 8 or more letters), average number of words in a sentence, and the sentence count. The same features are extracted from all the “perfect essays” provided to the system. The metrics are then normalized, and passed on for calculating variance with the test essay. The variance is calculated and a score is assigned based on the variance of different metrics.

Syntax Analysis Module

Ten different Part of Speech counts are computed for the test essay and each “perfect” essay individually. The Part-of-Speech include: Nouns, Verbs, Adjectives, Pronouns, Modal Auxiliary (“can”, “cannot”, “could”, “couldn’t”), Prepositions, or Conjunction, sub-ordinating conjunctions, Adverbs, Determiners, Coordinating Conjunctions (“neither”, “therefore”, “versus”, “whether”). These counts are then fed into a scoring function which normalizes them according to the word count for each essay. Again, the test essay is compared against the perfect essays and a score is computed and sent to the test program.

Spell Checker & Syntactic Error Checker

This module is in a passing build mode, and the ability to identify new types of errors are being continually added, in addition to improving the ones already implemented. Currently, the module can identify any miss-spelled words and a good variety of syntactic errors. The syntactic error checker works using a Context-Free Grammar using a recursive approach over the syntax tree.

The syntactic errors have been categorized into different “types”. type-1 errors include the errors where the sentence formation is incorrect or the sentence is incomplete. The same is detected using a recursive approach using Penn TreeBank and a handcrafted function that traverses over the tress. type-2 errors include miss-spelled words that come within some edit-distance such as “can’t’ vs “cant”.

Semantic Analysis Module

The module performs Latent Semantic Analysis over test essay and the set of “perfect” essays. Using a cosine spatial distance, it then assigns how the given test essay is apart from each of the “perfect essays”. The maximum value of cosine similarity is used to compute the score for this module. Thus, it would be in the best interests of the user to specify as diverse essays as possible in the set of “perfect-essays”.

Discourse Analysis Module

This module extracts key ideas from the essays. This is done using a context-free grammar looking for entities, key elements in Noun Phrases, Verb Phrases. It uses a regular-expression based parser and finds the similarity between key-ideas presented in the essay. The same is done using a unigram model, and a bigram model, interpolated according to a calibrated interpolation model, and a score is assigned.

Extra-credit implementation:

Weam thinking about another way to build the scoring system, which can be implemented by neural-network. For example, It is possible to implement a simple LSTM model with a scoring layer at the end. The LSTM takes words of an essay as time-steps, as it receives a stream of word vectors for the essay and computes a vector that captures the information present in the essay. The output layer converts this vector into a continuous score value.

Results:

Model type	Accuracy	RMSE
Linear Regression Model	0.804	0.407
KNN Model	0.692	0.510
Random Forest Model	0.798	0.413

We can see that the linear regression model works best on fitting scores generated from five modules to predict the final resolved essay scores.

Discussion and Conclusions:

Our goal was to create an automated essay grading system which can have a good performance in huge datasets. Our system is inspired by a modular design in (BURSTEIN and MARCU 2000) for building upon the current feature set by adding new textual cohesive devices which correspond to diverse characteristics of writing as perceived by a human grader. The final model clearly shows that modularity in such scoring systems help improve a lot. The current modular design includes syntactic variety , the similarity of ideas , correct usage and grammar, and organization of ideas at a very high level.

Because this system currently fits on essays in only one topic, which is persuasive type (Essay prompt: Write a letter to your local newspaper in which you state your opinion on the effects computers have on people). Wethink the scoring system might work better in some different type of literature works, such as expository essays. Because expository writing can have some rigorous requirement on wording and structure, which can improve the performance score in some modules.

Also, in the statistical modeling part, Weonly select the model with the best performance among linear regression, K-nearest neighbor and random forest models. It is possible that there are some other model which work better than Our current best linear regression one.

There is one unsolved challenge in building the Discourse Analysis module. It is tricky to find out ”flips”: Identify if the author has flipped tone within the essay at any point by partitioning essay into separate arguments. But Wethink it might be improved by using argument or topic distribution.