Automatic Argument Evaluation

Jason Liang and Eric Wang
Massachusetts Institute of Technology
{liangjy, wange}@mit.edu

Abstract

Our research project extends the work of the paper "Modeling Argument Strength in Student Essays" (Persing and Ng, 2015). Using tools provided by NLTK, Spacy, gensim, and Scikit-Learn, we train a SVM classifier to score argumentative and persuasive essays written by high schoolers based on the strength of the main argument is. In addition to features described in the original paper, such as POS n-grams and coreference features, we introduce a topic modeling component that attempts to capture the strength of the entire essay through emulating supports for claims. We also compare the accuracies of different classifiers, including SVMs and random forests.

1 Introduction

In recent years, automatic essay scoring (AES) has become one of the most controversial applications of natural language processing. Many critics (most notably Noam Chomsky), have voiced their doubts as to the validity and accuracy of any computer-based attempt to understand human writing. Their main argument is that computers and machine learning technologies have not yet, and may never, acquire enough knowledge to grasp the subtleties of writing. On that front, we cannot disagree with their assessment of the state of the art of AES systems.

A more specific critique is that current AES systems are unable to recognize a strong argument, and only use weak proxies for measuring argument strength such as argument length and word difficulty. However, "Modeling Argument Strength in Student Essays" (Persing and Ng, 2015) addresses this dimension by teaching a system how to score argumentative essays through examining more specific properties of a strong argument: cohesiveness, flow, coreference, etc.

The primary goals of this paper are threefold. First, we aim to replicate some of the findings described in the original paper of Persing and Ng, starting from baseline calculations up until and including POS n-grams, transitional phrases, and additional mentioned features.

Second, we will attempt to extend the methods and features used in the paper. Our primary additions are additional features for topic modeling and the use of random forests instead of SVMs. Topic modeling is traditionally used as a generative method to model patterns in a corpus of documents. We adapt it to argumentative essays to capture the multi-argument nature of good essays. Random forests are also tested as a method of regression in place of SVMs. Random forests are sometimes better at modeling non-linear relationships and can handle high-dimensional dta well.

Third, by extending the methods of Persing and Ng, we hope to develop a system that will come into widespread use. Especially with MOOCs (Massive Open Online Courses) becoming more popular, developing an algorithm that can accurately, consistently, and quickly score essays would allow for faster, cheaper, and more standardized grading. It would also greatly help students by granting them almost instantaneous feedback on their writing style and, even after the class, prepare them for the rest of their careers by giving them practice in argumentative and persuasive writing, an essential component of communication in any language.

2 Corpus Information

The corpus used in the original paper by Persing and Ng is the International Corpus of Learner English (ICLE): a collection of 1000 essays written by college-level students around the world

in response to several prompts (found at https://www.uclouvain.be/en-277586.html).

However, this dataset was not freely available, so we turned to a different dataset for our evaluation. Specifically, we used a dataset of essays written by middle and high schoolers provided by The Hewlett Foundation (https://www.kaggle.com/c/asap-aes/data). The essays in this corpus were written in response to eight prompts.

Out of these prompts, we chose to only use essays written by 10th graders in response to this prompt: "Write a persuasive essay to a newspaper reflecting your views on censorship in libraries. Do you believe that certain materials, such as books, music, movies, magazines, etc., should be removed from the shelves if they are found offensive? Support your position with convincing arguments from your own experience, observations, and/or reading". We chose this specific prompt since six of the other prompts did not ask students to write argumentative or persuasive essays. The responses to the remaining prompt were written by 8th graders, who we assumed would have a lower writing level than 10th graders.

The Kaggle dataset is a reasonable substitute for the ICLE corpus for several reasons. First, it contains a large number of argumentative essays. Because this project is primarily concerned with argument strength and not overall essay effectiveness, the essays themselves are sufficient as inputs to our learning algorithms. Second, the corpus provides grades purely based on the argument strength of the essays. Each essay is actually scored two ways: one score for how well it answers the prompt (which calls for persuasive writing) and supports its arguments; and one score how well it matches language conventions. We therefore know that the former score is exactly what we want, with no regard other possible factors for a "good" essay. Finally, each essay was scored by two human experts according to a detailed rubric, ensuring the accuracy of our training labels.

In addition, we obtained access to a corpus of TOEFL essays from the Linguistic Data Consortium (https://catalog.ldc.upenn.edu/LDC2014T06). Though we decided not to train our model on this dataset for this project, incorporating these TOEFL essays would be a natural extension to our work and would be extremely relevant to the development of industrial-grade automatic argument evaluators.

3 Score Prediction

In this section, we first describe the distribution of scores in our dataset. For each of the 1800 essays in the corpus, two different annotators scored the essay from 1-6 based on its argumentative merit, ideas, and content. Following are brief descriptions of the rubric for each score:

- **6**: presents a unifying theme or main idea without going off on tangents, stays completely focused on topic and task, includes in-depth information and exceptional supporting details that are fully developed, fully explores many facets of the topic
- **5**: presents a unifying theme or main idea without going off on tangents, stays focused on topic and task, provides in-depth information and more than adequate supporting details that are developed, explores many facets of the topic
- **4**: presents a unifying theme or main idea (writing may include minor tangents), stays mostly focused on topic and task, includes sufficient information and supporting details (details may not be fully developed, ideas may be listed), explores some facets of the topic
- **3**: attempts a unifying theme or main idea, stays somewhat focused on topic and task, includes some information with only a few details or lists ideas without supporting details, explores some facets of the topic
- **2**: attempts a main idea, sometimes loses focus or ineffectively displays focus, includes little information and few or no details, explores only one or two facets of the topic
- **1**: difficult for the reader to discern the main idea, too brief or too repetitive to establish or maintain a focus, include littles information with few or no details or unrelated details, is unsuccessful in attempts to explore any facets of the prompt

Score	1	2	3	4	5	6
Essays	48	308	1487	1594	148	15

Table 1: Scores of essays in corpus

Having two graders score each essay allows us to define a gold standard for the accuracy of our model which is based on annotator agreement.

There are 1800 essays in total, out of which there are 1410 (78%) on which the graders exactly agree and 1796 (99.8%) on which the scores of the two graders fall within 1 point of each other.

From the score distribution, we can see that we have a skewed distribution of classes. There are many more intermediate scores (3s and 4s) than extreme scores. This raises questions about what metric we want to use to score our models. Do we want the average error (absolute difference between predicted and actual scores) to be low, or do we care about classifying extreme examples correctly?

4 Baseline Systems

To begin our analysis, we first implemented two baseline systems. The first baseline is as follows: let the most common score in the training set be s. Then, we assign to each essay in the test set the score s. From Table 1, we see that s will most likely be 3 or 4. We call this the "most frequent" baseline.

The second baseline is based on the presence of discourse connectives in the essay. These are words such as once, since, and on the contrary that describe how two segments of discourse are logically connected to one another. There are four main classes of discourse connectives:

- expansion: one clause is elaborating information in the other
- comparison: information in the two clauses is compared or contrasted
- contingency: one clause expresses the cause of the other
- temporal: information in two clauses are related because of their timing

We use heuristic rules developed by Ong et. al. and also used by Persing. These rules are:

- Whether the essay contains at least one sentence labeled hypothesis (contain string prefixes from "hypothes" or "predict" but do not contain string prefixes from "conflict" or "oppose", or contain the word "should" and contain no contingency connectives or string prefixes from "conflict" or "oppose")
- Whether the essay contains at least one sentence labeled opposes (begins with a comparison discourse connective or contains any string prefixes from "conflict" or "oppose")
- The sum of claim (contains any string prefixes from "suggest", "evidence", "shows", "essentially", or "indicate") and supports (begins with a contingency connective) sentences divided by the number of paragraphs in the essay

Due to the format of the discourse connectives tagger which was provided (http://www.cis.upenn.edu/~nlp/software/discourse.html), we resorted to using pattern matching to label different discourse connectives. In the future, we plan to use the more advanced parse-tree based tagger to identify discourse connectives or to train our own tagger using the Penn Discourse Treebank https://www.seas.upenn.edu/~pdtb/.

5 Our Approach

This section presents descriptions of the additional features and models that were used beyond our implementation of baseline 2.

5.1 POS N-grams

The first feature class that we add to the baseline is POS n-grams. In particular, we have one feature for every possible sequence of 1-3 POS tags.

We used the Spacy NLP library (https://spacy.io/) to annotate the corpus with POS tags, since Spacy is much faster than NLTK. We then count the number of times each sequence appears in the essay and normalize this vector to unit length.

5.2 Semantic Frames

As their second additional feature class, the paper of Persing and Ng uses semantic frame features. Semantic frames are descriptions of types of events, relations, and entities and the associated participants. For example, the act of eating involves the person who is eating, the food that is being eaten, and the utensil which is being used to eat. Thus, a feature which can be extracted is "Eat-Utensil-Fork".

We were not able to find a Python library that could produce semantic frame annotations. While NLTK (http://www.nltk.org/howto/framenet.html) provides a library for browsing semantic frames, it does not yet have an API for annotating sentences with semantic frame features. Thus, we decided not to include this feature in our model.

5.3 Transitional Phrases

We used the 14 transitional phrase lists found at http://www.studygs.net/wrtstr6.htm to produce additional features. Specifically, for each phrase list, a feature was created which counted the average number of transitions from the list that occurred per sentence. These features may be able to help score the "flow" of the essay and how the essay transitions between ideas.

5.4 Coreference

Similarly to the authors of the original paper, we augment the model with coreference features, which are described on the website http://www.hlt.utdallas.edu/~persingq/ICLE/asCorefFeatures.txt. Counting coreferences is another method of measuring how unified the theme of the essay is.

Introducing these features was especially simple since the corpus downloaded from Kaggle already contains anonymized named entities. For example, if the original sentences are "John likes to eat apples. John also likes to cook.", the sentences as they appear in the corpus would be "@PERSON1, likes to eat apples. @PERSON1 also likes to cook."

5.5 Other Features in Persing and Ng

Persing and Ng introduce other features which include prompt agreement, argument component predictions, and argument errors (which includes features for the number of sentences in the essay, whether paragraphs have major claims, and whether paragraphs have argument components). We opted not to use these features because they would have required us to annotate all the essays in the corpus with the needed information.

5.6 Topic Models

As the first of our novel methods, we decided to incorporate Latent Dirichlet allocation (LDA) into our model. LDA is a type of topic model which generates a document given a distribution of topics and given a distribution of words in a topic.

LDA differs from other topic models in that the topic distribution in each essay is modeled as a mixture of global topic distributions. The LDA model can be expressed as

$$p(w_1, \ldots, w_N|\Theta) = \sum_{j=1}^{M} p_j [\prod_{i=1}^{N} \sum_{z} \Theta_z^j \Theta_{w_i|z}]$$

Where the w_i are the words in the essay, there are N words, there are M global topic distributions, Θ_j^z is the jth global topic distribution, z ranges over all topics, and p_j is the probability that the document's topics are generated from the jth topic distribution.

To break it down, we provide a step-by-step explanation of this calculation. First, to calculate the probability that word i appears, we take the sum of the probabilities of the word being generated by each topic. We then multiply over all N words to get the probability that the sequence of words was produced by the jth topic distribution. Lastly, we sum over all M topic distributions to get the final probability that an essay is generated.

We introduce six new features to our model based on LDA. We group essays in the training set with the same score together. We use each of these groups to train a LDA topic model with one topic. This generates six different topic models. For each essay in the test set, we compute the log likelihood that the essay was generated by the topic model. This results in six log likelihood values, which are added as features to our model.

5.7 Random Forests

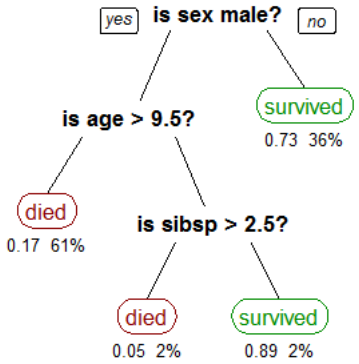
Random forests are an alternative classification scheme to using SVCs. They are closely related to decision trees, which provide a tree-like structure for performing classification. They can also be adapted for regression, which is what we use.

There are several different algorithms to train a decision tree. The most widely used is ID3, which begins with the entire training set at the root node. On each iteration of ID3, the attribute of the training set with the highest information

gain is chosen as the next decision branch. This continues until all training examples have been classified correctly.

To train a random forest classifier, bootstrap aggregating is used. Suppose we want to have B decision trees in our random forest, and suppose that there are n examples in the training set. To build each decision tree in the random forest, we randomly choose n training examples (with replacement) from the training set, and build a decision tree with those training examples. Typically, hundreds to thousands of trees are made. In addition, supposing that there are p features in the original training set, we should train the decision tree using $\sqrt{p}$ features. This is called feature bagging. The prediction for each test essay is the average of the predictions for each essay in the random forest.

Training using many decision trees decreases the variance of the model, since the model is less affected by noise in the training data. Also, we use fewer features to train each tree in the random forest to reduce correlation of the trees: if one feature is a very good predictor, this feature will be used in many decision trees.



6 Evaluation

6.1 Metrics

To compare our method (LDA + random forest classifier) with baselines 1 and 2, and Persing and Ng, we used mean squared error between the predicted and actual score. This is different from Persing and Ng, in which they use percentage of scores correct, mean error, and the aforementioned metric. We do this for several reasons.

First, we are formulating this as a regression problem, so percentage accuracy is not appropriate. We also want to penalize vastly different scores more harshly, so we use mean squared error over mean absolute error. Other metrics that we can use in the future are Pearson correlation coefficient, which measures how well the regressor can fit the data.

6.2 Results and Discussion

The table below gives the results of our baselines, as well as implementations of Persing and Ng's method. We used the aforementioned corpus, cross-validated 5 times to ensure stability.

System	Mean squared error
Baseline 1	0.93
Baseline 2	0.78
Persing and Ng	0.60
Our model	0.59

Table 1: Scores of essays in corpus

Our system performs slightly better than our implementation of some of Persing and Ng features. This can be due to several reasons apart from a better model. The first is that we used a different data set. It is possible that our model better emulates 10th grade essays, which are less complex on average than the ICLE essays used by Persing and Ng (less complex in terms of argument, but possibly better English convention). Furthermore, a different of 0.01 is very small and not likely to be a significant improvement to the model given by Persing and Ng. However, this does mean that our model proves to be a reasonable alternative with features that may be useful for future exploration.

7 Conclusion

Using a dataset obtained from Kaggle, we attempt to replicate the analysis of the paper "Modeling Argument Strength in Student Essays" by Persing and Ng. In addition to implementing the most frequent and discourse connectives baselines, we also include the POS n-gram, transitional phrase, and coreference feature classes described in the paper. On of our major contributions to the model is introducing LDA topic models in order to measure how cohesive an essay's arguments are. Our other

major addition was using a random forest classifier instead of a SVM classifier.

References

Github repository: https://github.mit.edu/liangjy/essay-scoring

Nathan Ong, Diane Litman, and Alexandra Brusilovsky. 2014. Ontology-Based Argument Mining and Automatic Essay Scoring. https://www.aclweb.org/anthology/W/W14/W14-2104.pdf.

Vincent Ng and Isaac Persing. 2015. Modeling Argument Strength in Student Essays. http://aclweb.org/anthology/P15-1053.

Ani Nenkova and Emily Pitler. 2009. Using Syntax to Disambiguate Explicit Discourse Connectives in Tex. http://www.aclweb.org/anthology/P09-2004.

ETS paper: https://www.ets.org/Media/Research/pdf/RR-04-45.pdf