Midway report

Parth Tamane
ptamane3@gatech.edu

Gaurav Pande
gpande@gatech.edu

Naila Fatima
nfatima3@gatech.edu

Abstract

Automated essay scoring (AES) is an important area of natural language processing. Our goal is to explore different techniques in machine learning and deep learning to automatically score essays and identify what works best. We also want to understand if any bias affects the score assigned to an essay. So far we have implemented linear regression based model on style based features, basic LSTM model with word2vec embedding and gender prediction models using sentiment and content based features. Going forward we would like to explore Word2Vec and LSA based features for essays, build better deep learning architecture using other embeddings like BERT and build age prediction models.

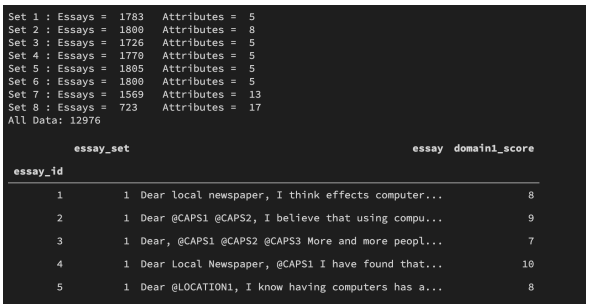
1 Goal

We aim to train models that grade in a manner similar to human graders. We also aim to develop models which can predict author demographic information from the text and this information will be used to analyze whether there is a relationship between the essay scores and the author demographic. Developing these models will allow us to understand how essays are graded by humans and may help us understand how essay grading can be further improved. Our goal is still the same and we are on track.

2 Progress Made

2.1 Data

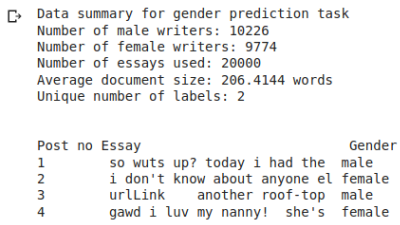
We have used the The Hewlett Foundation: Automated Essay Scoring data set.(Hewlett, 2012: accessed March 12, 2020) The size of the data set is 186.3 MB. It contains a file with essays for 8 sets and scores assigned by multiple graders (domain1_score is defined for all essay sets). The (min, max) scores for sets are: 1 - (2, 12), 2 - (1, 6), 3 - (0, 3), 4 - (0, 3), 5 - (0, 4), 6 - (0, 4), 7 - (0, 30), 8 - (0, 60). Along with this, the data set also contains files explaining the grading scheme and essay prompts for the 8 sets.



```
Set 1 : Essays =  1783   Attributes =  5
Set 2 : Essays =  1800   Attributes =  8
Set 3 : Essays =  1726   Attributes =  5
Set 4 : Essays =  1770   Attributes =  5
Set 5 : Essays =  1805   Attributes =  5
Set 6 : Essays =  1800   Attributes =  5
Set 7 : Essays =  1569   Attributes =  13
Set 8 : Essays =  723    Attributes =  17
All Data: 12976
```

essay_id	essay_set	essay	domain1_score
1	1	Dear local newspaper, I think effects computer...	8
2	1	Dear @CAPS1 @CAPS2, I believe that using compu...	9
3	1	Dear, @CAPS1 @CAPS2 @CAPS3 More and more peopl...	7
4	1	Dear Local Newspaper, @CAPS1 I have found that...	10
5	1	Dear @LOCATION1, I know having computers has a...	8

Figure 1: Data Set Summary for Automated Essay Scoring [Null columns dropped]



```
Data summary for gender prediction task
Number of male writers: 10226
Number of female writers: 9774
Number of essays used: 20000
Average document size: 206.4144 words
Unique number of labels: 2
```

Post no	Essay	Gender
1	so wuts up? today i had the	male
2	i don't know about anyone el	female
3	urlLink another roof-top	male
4	gawd i luv my nanny! she's	female

Figure 2: Data Set Summary for Gender Prediction Task

There are 2 distinct types of essay prompts and each set has one type of prompt: 1) Persuasive / Narrative / Expository: Set 1, 2, 7, 8 and 2) Source dependent responses: Set 3, 4, 5, 6.

For the gender prediction task, we have used the 'Blog Posts labeled with Age and Gender' dataset obtained from Kaggle. (J. Schler and Pennebaker, 2004: accessed March 13, 2020) This dataset has around 530,000 blog posts (630.2 MB) annotated with the blogger's age and gender. Since it is a massive dataset, it caused RAM problems on Colab and we decided to use the first 20,000 blog posts as our data.

2.2 Method

In order to build our grading model, we have used both machine learning and deep learning based methods. Our regression based model relies on style based features. Prior work has used regression with other classifiers like KNN and Naive Bayes to create a sort of ensemble.(Larkey, 1998) We are trying to improve the performance by using only regression and identifying features that are indicators of a well written essay. This gives us good results.

Our deep learning based model used word2vec embedding which were fed to 2 LSTM layers, the output of which is then fed to a linear softmax layer. We are using mean square error as loss function and rmsprop as an optimizer. We used RELU as activation unit with dropout value of 0.5. This is fundamentally same as prior work done by Kaveh Taghipour and Hwee Tou Ng (Taghipour and Ng, 2016), but we are planning to improve it by using different embedding like BERT.

For the gender prediction task, we have experimented with POS (part-of-speech) features, POS n-grams, n-grams, possessives and sentiment features. Most of the papers that we have read use POS and n-gram features with logistic regression and SVMs (support vector machines), but we have tried to use several other features. Our models with possessive and sentiment features did not work as well as the traditionally used n-gram and POS features. We have tested our models on our dataset by measuring the accuracy of the predictions made by the model. We are seeing good results on our dataset (around 64% accuracy) and are trying to improve it further.

2.3 Preliminary Results

We have implemented linear regression to predict the scores that would be assigned to an essay. We use 5-fold cross validation to compute the mean quadratic kappa score, which gives the correlation between 2 raters grading. Our initial tests are giving good results with a mean quadratic kappa score (our success metric) of 0.822 using the entire dataset. This is a pretty good score. We had experimented with SVMs and KNN models too. But even with best hyperparameters, SVM gave a mean quadratic kappa score of 0.8103 and KNN gave 0.7953 which was easily surpassed by linear regression giving 0.8422 (figures reported for set 1). We tested for a single set in the interest of time. Figure 3 shows the scores assigned to different essays in set 1.

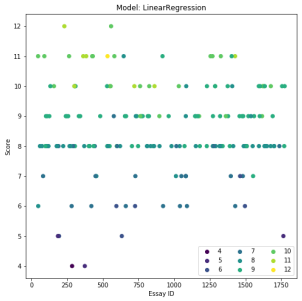


Figure 3: Set 1 Essay Score Predictions

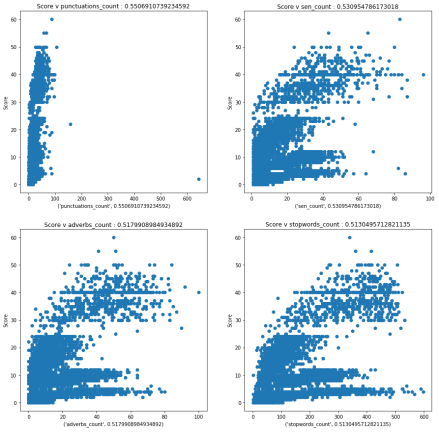


Figure 4: Individual Features with top Kappa Score

The mean quadratic kappa score was achieved using all the 8 essay sets after we added 8 new style based features. These were:

1) spelling error count, 2) stop words count, 3) small sentences count (less than 4 words), 4) punctuation count, 5) verbs count, 6) adverbs count, 7) nouns count, 8) adjectives count.

These additional features helped bump the score up from 0.747 when we used the basic 11 features. (Larkey, 1998) Our intuition that more style based features will help capture more information about the essay sets turned out to be true. In figure 4, we can see that out of the top 4 features, 3 were the ones we added. This graph also gives a good idea of how these different features are distributed with scores for essays across different sets.

Our hypothesis that content based techniques work better on less open ended prompts while style based techniques work better on more open ended prompts was shown to be correct.

You can see in table 1, and table 2, for all open ended Type 1 essays, the Kappa score went up as

Set Id	1	2	3	4
11 Feat.	0.8388	0.6912	0.6438	0.6888
All Feat.	0.8422	0.6977	0.6458	0.6879

Table 1: Mean Quadratic Kappa Scores per Set - I

Set Id	5	6	7	8
11 Feat.	0.7777	0.6626	0.7303	0.7241
All Feat.	0.7753	0.6778	0.7703	0.7266

Table 2: Mean Quadratic Kappa Scores per Set - II

more style based features were added. But this did not help with Type 2 essays. The quadratic mean score actually slightly went down with newer features.

In deep learning side, the first task is to experiment with different embeddings and see if they yield any better results. So far we have implemented word2vec embeddings and tried to understand what the data represents and what can be most common pitfalls while using the embeddings. We built the word2vec embeddings using 300 features, and 100 minimum words for embedding, and we tried to visualize similarities between different words in the data set. Since the data is in higher dimensional space, we used t-sne (Laurens van der Maaten, 1970) for calculating the similarities and visualizing them as shown in figure 5. Based on this visualization, we can see that the dataset contains many similar words which have spelling mistakes, and also it captures the semantic words as well which are meant for tagging different words in the essay. Based on this, the model didn't exceed more than 60 mean quadratic kappa score. The model wasn't successful because we were not able to achieve the desired performance in mean quadratic kappa score.

For the gender prediction task, we used features such as possessives, part-of speech (POS) counts (Yatam SS, 2014), POS bigrams, f-measure (Mukherjee and Liu, 2010), n-grams (Rao et al., 2010), (Yatam SS, 2014) and sentiment features. Feature extraction was performed after applying tokenization (and lemmatization for sentiment features) to the data. The models used to perform the classification were support vector machines (SVMs), k nearest neighbors (kNN), Naive Bayes and logistic regression. Several papers that we have read for this task focus on predicting the gender of a social media (generally Twitter) user. The use of POS and n-gram features generally gives

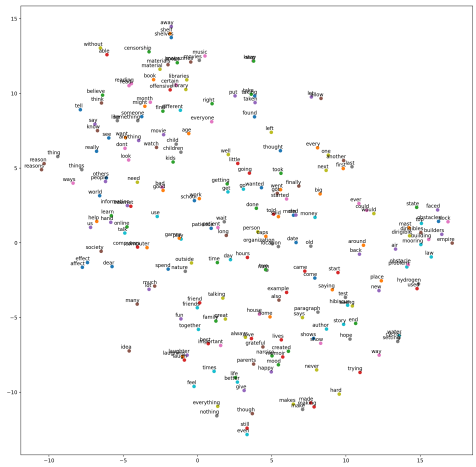


Figure 5: Word similarities for overall dataset using t-SNE

an accuracy between 50-60% (our baseline model: (Yatam SS, 2014)) but several papers use username information (as several social media users have included their name in their username), profile colors, use of emoticons and profile photos to improve the performance of their model. Since the HP dataset does not contain any personally identifying information, we have focused on predicting the gender of an author from their written work only. As essays contain formal language, we cannot use emoticons, acronyms (such as LOL) and address terms (such as 'bro') to aid our prediction task. Due to this, our accuracies may not be as high as those reported in the papers. As seen in Figure 8, we achieved our best performance on using n-gram features with a Naive Bayes model. Figure 6 shows the difference between the n-gram word-level features for both genders. Each graph shows the plot between the number of counts of an n-gram (Y-axis) against the n-gram (X-axis) for a particular gender. Our next best performance was achieved by using POS count features. Figure 7 shows the POS-count features for both genders. From Figure 8, we can observe that SVMs (with Gaussian kernel) tend to perform better than other models for a particular feature. The sentiment, f-measure and possessive features didn't work as well as expected. This may have occurred since f-measure is too dependent on a linear combination of POS tags and social media users (irrespective of their gender) express several different sentiments online.

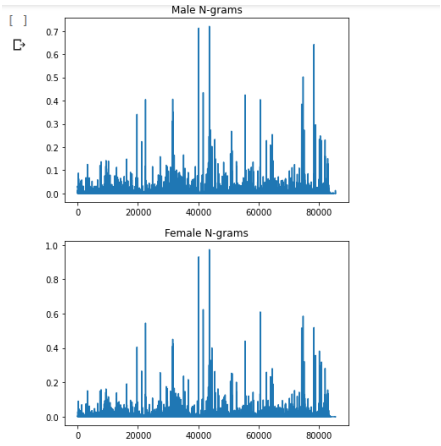


Figure 6: N-gram (word-level) features - males and females

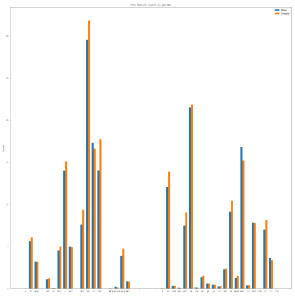


Figure 7: POS-counts features for males and females

Feature used	Accuracy with best model
Possessives	kNN model: 52.45 (k=75)
POS counts	SVM: 60.375
POS bigrams	SVM: 59.75
N-grams (unigram, word level)	Naive Bayes: **64.05**
Sentiment	SVM: 52.15

Figure 8: Accuracy for models used

2.4 Work Division

1) Parth was responsible for extracting the style based features. He also evaluated different models like linear regression, SVM and KNN.
2) Gaurav worked on word2vec embeddings and exploration of deep learning architecture to have better result. So far, word2vec embeddings and basic LSTM model were experimented with.
3) Naila developed the gender prediction model by extracting various style, content and sentiment based features. She also trained several different classification models such as logistic regression, Naive Bayes, SVMs and kNNs.

3 Plan to complete the project

3.1 Future tasks

What are the next steps in your project, both in the short-term and the long-term?

The long term goal of the project is to build a model which has higher correlation with human graders. We also want to understand how biases affect the score given by human graders. Our short term goals we will be:
1) Generating content based features to improve quadratic mean kappa score by calculating the similarity between the required reading and the written essays (Islam and Hoque, 2010) and generating word vector representation of essays.
2) Trying different deep learning architecture and using different embeddings like BERT embeddings (Devlin et al., 2018) to improve the kappa score.
3) Developing an age prediction model and using these 2 models to find underlying biases, if any.

If you are developing a model, what is the next version of the model? What results do you expect from the next model and why?

1) We will be incorporating content based features to improve the correlation score for type 2 essays using regression. Similarity measures with prompt reading content and word vector essay representations will be used. These additional features will help increase the kappa score as they will capture more information about essay content.
2) In Deep Learning we will try BiLSTM model with BERT or Glove embedding which will help in improving the train and validation kappa score as they will capture more features based on attention.

3) We will be creating an age prediction model by experimenting with n-grams, possessives among other features. We will try to use SVMs, linear regression and kNNs to develop a good model. We will try to use word embeddings to improve our model.

If you are analyzing results, what tests do you plan to run on your data? How much more testing will you require to answer your original research question? What results do you expect?

In order to answer our original research question, we will use our age and gender prediction models and score estimator to find if a correlation exists between the demographic of an essay writer and the score allotted to their writing. We are unsure of whether we will detect a bias in essay grading as this is a grader specific phenomenon.

What is the timeline for the future tasks?

1) We will first explore content based feature extraction. After that word vectors will be used to represent essays and a final analysis will be done to see how all the features affect the kappa score.
2) In Deep Learning based model, next week will be spent on understanding the BERT model, then for the week after that, we focus on its implementation using BiLSTM as base model.
3) Within a week, we will try to extract more content-based features and use embeddings. After that, we will build models to perform age prediction.

3.2 Work division

Which team members will be responsible for which remaining parts of the project?

1) Parth will be working on generating content based features from the prompts for essay set 3-6. He will also be working on integrating word vectors with the other 19 features extracted so far.
2) Gaurav will be working on Deep learning model and embeddings like BERT or Glove.
3) Naila will be working improving the gender prediction model and developing the age prediction model. She will then try to analyze the correlation between author demographics and scores.

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding.

Hewlett. 2012: accessed March 12, 2020. *The Hewlett Foundation: Automated Essay Scoring*.

M. Monjurul Islam and A. S. M. Latiful Hoque. 2010. Automated essay scoring using generalized latent semantic analysis. In *2010 13th International Conference on Computer and Information Technology (ICCIT)*, pages 358–363.

S. Argamon J. Schler, M. Koppel and J. Pennebaker. 2004: accessed March 13, 2020. *Blog Posts Labeled with Age and Gender*.

Leah S. Larkey. 1998. Automatic essay grading using text categorization techniques. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, page 90–95, New York, NY, USA. Association for Computing Machinery.

Geoffrey Hinton Laurens van der Maaten. 1970. Visualizing data using t-sne.

Arjun Mukherjee and Bing Liu. 2010. Improving gender classification of blog authors. *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*.

Delip Rao, David Yarowsky, Abhishek Shreevats, and Manaswi Gupta. 2010. Classifying latent user attributes in twitter. *Proceedings of the 2nd International Workshop on Search and Mining User-Generated Contents - SMUC 10*.

Kaveh Taghipour and Hwee Tou Ng. 2016. A neural approach to automated essay scoring. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*.

Reddy TR Yatam SS. 2014. Author profiling: Predicting gender and age from blogs, reviews social media. *Int J Eng Res Technol 3:631–633*.