N-gram Based Approach for Automatic Prediction of Essay Rubric Marks

Magdalena Jankowska, Colin Conrad, Jabez Harris, and Vlado Kešelj

Faculty of Computer Science, Dalhousie University, Halifax NS, Canada
magdalena.jankowska@dal.ca, colin.conrad@dal.ca, jay.harris@dal.ca,
vlado.keselj@dal.ca

Abstract. Automatic Essay Scoring, applied to the prediction of grades for dimensions of a scoring rubric, can provide automatic detailed feedback on students' written assignments. We apply a character and word n-gram based technique proposed originally for authorship identification—Common N-Gram (CNG) classifier—-to this task. We report promising results for the rubric mark prediction for essays by CNG, and perform analysis of suitability of different types of n-grams for the task.

Keywords: Automatic Essay Scoring, text classification, character n-grams

1 Introduction

Essay writing is an important component of formal education. In order to improve their writing, students require effective and specific feedback on different writing dimensions. Automated Essay Scoring (AES) is an ongoing area of research that seeks to expedite the feedback process. Human-generated scores are often treated as a gold standard and automated tools are built to learn from and predict these scores [1]. Recently, the subject has received renewed attention, due to the Automated Student Assessment Prize (ASAP) Competition data[1] [2].

Detailed rubrics are often used to give specific qualitative feedback about ways of improving writing, based among others on stylistic elements contained in the text itself. The task of rubric prediction can be treated as a way of generating detailed feedback for students on the different elements of writing, based on the characteristics typical of a student's performance. This is fundamentally similar to the task of the automatic identification of an author of a text. In this paper, we explore the application of the Common N-Gram (CNG) classifier [3], which is frequently used in authorship analysis, to the task of rubric value prediction. Using the ASAP data, we compare performance of the CNG algorithm to two other, popular text classifiers: linear Support Vector Machines (SVM) with stochastic gradient descent (SGD) learning and Multinomial Naïve Bayes algorithm (NB). We also compare the results to the reference of the inter-rater agreement between human raters.

[1] https://www.kaggle.com/c/asap-aes

2 Related Work

Automated Essay Scoring is not a new concept. The earliest AES systems date back to the 1960's. One of the most notable commercial AES systems is e-Rater [4], which is used to evaluate second-language English capabilities on the Test of English as a Foreign Language (TOEFL) and essay rating of the Graduate Record Examination (GRE). Other innovations such as Gradescope [5] are now enabling a transformation in education delivery. Much of the work on AES treats automated grading as a supervised learning problem, using features related to content (e.g., through latent semantic analysis [6]) or style. The e-Rater system, for instance, performs feature extraction related to specific writing domains such as *grammar*, *usage*, *mechanics* and *style*. This paper likewise treats AES as a supervised classification problem.

The Common N-Gram classifier (CNG) technique explored in this paper is conventionally used in the context of authorship attribution, which is the task of detecting who among candidate authors wrote a considered text. CNG is based on a similarity measure between documents relying on differences between frequencies of the character n-grams. The CNG similarity, or its variants, has been successfully applied to tasks related to authorship analysis of texts and related applications [7, 8]. It has also been explored in the context of AES [9] but has not been evaluated using a popular dataset.

3 Methodology

Prediction of the rubric grade is performed using supervised classification and is performed separately for each detailed rubric dimension, such as "Style", "Organization", etc., with classifiers trained using marks given by human raters. We applied three classification algorithms: the CNG classifier, linear SVM with SGD learning, and NB. The representation of documents is based on n-grams of characters or words. We also tested "stemmed word" n-grams.

A representation of a document used by CNG is a "profile"—a list of the most frequent n-grams of a particular type, coupled with their frequency normalized by the text length. The total number of unique n-grams in a document was set as the length of a profile. Training data for a class is represented by CNG as a single class document, created by concatenating all training documents from the class. For SVM and NB, we used a typical bag-of-n-grams representation (using a particular n-gram type as features), with either raw counts or *tfidf* (term frequency—invert document frequency) scores as weights. To mitigate the effect of unbalanced training data for SVM and NB, we experimented with random upsampling of minority classes. CNG is known to be especially sensitive to unbalanced training data [10], but its nature prevents the use of upsampling of training documents; for CNG we addressed the problem by truncating all class profiles to the same, maximum possible, length.

4 Experiments

4.1 Data

Experiments were performed on essays of three ASAP datasets: set 2, set 7 and set 8. These three sets of essays are chosen for experiments (out of eight sets available in the dataset), because for these sets marks assigned to individual dimensions of the evaluation guideline rubric (such as "Style", "Organization", etc.) are available. Table 1 presents information about the essay sets.

Table 1. Information about ASAP datasets used in experiments

name	set2	set7	set8
grade level	10th	7th	10th
# of essays	1800	1569	723
average # of words	381	168	606
rubric dimensions	"Writing Applications" "Language Conventions"	"Ideas" "Organization" "Style" "Conventions"	"Ideas and Content" "Organization" "Voice" "Word Choice" "Sentence Fluency" "Conventions"

For each rubric dimension, grades from two raters are available. Classification is performed for each dimension and each rater separately, and so there are 24 classification tasks in total. The number of classes (different marks) is 4 for all sets and dimensions except for "Writing Applications" in set 2, in which the number of classes is 6. For set 8, our classification is for 4 classes, but the original scale of marks has 6 marks: from 1 to 6. We combined for this set mark 1 with mark 2, and mark 6 with mark 5 because the extreme marks are very rare.

4.2 Experimental Settings

We performed experiments for 13 feature sets: character n-grams of the length from 2 to 10, and word and stemmed word n-grams of the length of 1 and 2. For each of 13 feature sets, one classification by CNG was performed while for SVM and NB each, four classifications were performed, for the combinations of two types of n-gram weights and two types of processing of unbalanced training data (upsampling and no upsampling). The performance measure used in the experiments is Quadratic Weighted Kappa (QWK). Testing was performed using 5-fold stratified cross-validation. For SVM with SGD learning and NB we used implementations of the classifiers from Scikit-learn Python library [11]. Processing of documents in order to extract "stemmed word" n-grams was performed using Snowball Stemmer and the English stop words corpus from the NLTK platform [12]. The package imbalanced-learn [13] was utilized to perform the upsampling of training data.

4.3 Results and Discussion

Table 2. Best QWK results among all trained classifiers for each task

inter-rater	rater1			rater2		
QWK	best classifier	QWK	diff	best classifier	QWK	diff
			set2			
0.814	SVM tfidf upsam. char6	$0.567_{\ddagger}$	30%	SVM tfidf upsam. char6	$0.571^{*}_{\ddagger}$	30%
0.802	SVM tfidf upsam. char5	0.570	29%	SVM tfidf upsam. char4	0.567	29%
			set7			
0.695	CNG char4	0.657	6%	NB counts char3	0.628	10%
0.577	SVM tfidf upsam. char6	0.508^{*}	12%	SVM tfidf upsam. char6	0.515^{*}	11%
0.544	SVM tfidf upsam. char5	0.480	12%	SVM tfidf upsam. char5	0.493^{*}	9%
0.567	SVM tfidf upsam. char4	$0.428_{\ddagger}$	25%	SVM tfidf upsam. char5	$0.486^{*}_{\ddagger}$	14%
			set8			
0.523	NB counts upsam. char3	$0.482_{\dagger}$	8%	CNG char5	0.394	25%
0.533	NB counts char3	$0.455_{\dagger}$	15%	CNG char5	0.377	29%
0.456	CNG char4	0.440	4%	CNG word1	0.377	17%
0.477	CNG char4	0.493	-3%	CNG char5	$0.431_{\dagger}$	10%
0.498	CNG char4	0.489	2%	CNG char4	$0.459_{\dagger}$	8%
0.532	CNG char4	$0.454_{\dagger}$	15%	CNG char4	$0.436_{\dagger}$	18%

Table 2 presents the best result among all trained classifiers for each classification task (a row corresponds to a rubric dimension). For each rubric dimension, the QWK of the inter-rater agreement between rater1 and rater2 is also stated as a reference. "diff" is a relative difference between an inter-rater QWK and a given classifier QWK, as a percentage of the inter-rater QWK. Classifier QWK values denoted by the symbol * (respectively $_{\dagger}$, $_{\ddagger}$) are statistically significantly higher ($p < 0.05$) than the best in the task result of the CNG classifier (respectively SVM with tfidf and upsampling, NB with counts and upsampling).

We can observe that the algorithms which achieved the best overall results for a given task were CNG (11 tasks), SVM with tfidf scores and upsampling of training data (10 tasks) and NB with counts (3 tasks). It can be noted that the best results of classifiers often do not differ in a statistically significant way.[2] Finally, we can note that the best performance achieved on particular classification tasks varies substantially when compared to the agreement of the human raters. Set 2 demonstrates the highest agreement between the human raters, and demonstrates the highest discrepancy between the raters and the classifiers. On set 8, by contrast, the results of classifiers are relatively close to the inter-raters QWK (especially for rater1, on three dimensions, the QWK of CNG differs from the inter-rater QWK by less than 5%). These results indicate that CNG is a promising method for the problem.

[2] A correction for multiple hypothesis testing was not applied; it would be valuable if outperforming performance of any classifier was to be established.

Table 3. Six best performing features for selected classifiers

CNG		SVM tfidf upsam.		NB counts no samp.		NB counts upsam.	
feature	"#good"	feature	"#good"	feature	"#good"	feature	"#good"
char 4	24	**char 4**	20	**char 3**	22	**char 3**	23
char 5	23	**char 5**	18	**char 2**	19	**char 4**	19
char 6	19	**char 3**	18	char 4	7	**word 1**	18
word 1	12	**char 6**	17	word 1	3	**char 2**	17
char 7	11	**word 1**	13	stem 1	3	**stem 1**	12
char 8	9	**stem 1**	12	char 5	2	char 5	10

We performed feature analysis for four selected classifiers. For each of the classifiers, we ranked the n-gram types by a number of tasks (out of 24), in which a given type was not statistically significantly worse than the best performing n-gram type in the task—we called this number "#good". In Table 4.3, we report six best feature sets for each classifier, and denote as bold the ones for which "#good" is greater or equal to the half of the total number of tasks.

The analysis indicates that character 4-grams and 5-grams are the best features for CNG and SVM; character 6-grams and word unigrams are also well suited for these two classifiers. Character 3-grams perform well for SVM, and short character n-grams of the length 2, 3 and 4 perform especially well for NB.

5 Conclusion and Future Work

Promising results were obtained for rubric prediction based on character n-grams and word unigram representations, using CNG classifier, SVM with SGD learning with tfidf scores, and Naïve Bayes with raw counts, when compared to the inter-rater agreement between the scores of human raters. CNG algorithm, proposed originally for author identification, performed well compared to the other classifiers and shows promise for future study. Several methods of improving the performance of the prediction could be investigated in future work. They include combining different types of n-grams, either by using them together in document representation, or by an ensemble of classifiers based on different n-grams, as well as combining n-gram-based features with other types of features, such as parts of speech, detected spelling/grammar errors or presence of prescribed words. It would be worthwhile to investigate the relationship between the classifier performance and the type of rubric dimensions, as well as the impact of the level of inter-rater agreement. Future research could focus on investigating the role CNG and its similarity can play in complementing existing processes in AES tasks.

6 Acknowledgement

The project was supported by the NSERC Engage grant EGP/507291-2016 with industry partner, D2L Corporation. The authors would like to thank D2L mem-

bers: Brian Cepuran, VP, D2L Labs and Rose Kocher, Director, Grant & Research Programs, for their guidance in the project and the feedback on the paper. The authors would also like to acknowledge a support from Killam Predoctoral Scholarship.

References

1. Shermis, M.D., Burstein, J.: Handbook of automated essay evaluation: Current applications and new directions. Routledge (2013)
2. Phandi, P., Chai, K.M.A., Ng, H.T.: Flexible domain adaptation for automated essay scoring using correlated linear regression. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. (2015) 431–439
3. Kešelj, V., Peng, F., Cercone, N., Thomas, C.: N-gram-based author profiles for authorship attribution. In: Proceedings of the Conference Pacific Association for Computational Linguistics, PACLING'03, Dalhousie University, Halifax, Nova Scotia, Canada (August 2003) 255–264
4. Attali, Y., Burstein, J.: Automated essay scoring with e-rater® v. 2.0. ETS Research Report Series **2004**(2) (2004)
5. Singh, A., Karayev, S., Gutowski, K., Abbeel, P.: Gradescope: A fast, flexible, and fair system for scalable assessment of handwritten work. In: Proceedings of the Fourth (2017) ACM Conference on Learning@ Scale, ACM (2017) 81–88
6. Foltz, P.W., Laham, D., Landauer, T.K.: Automated essay scoring: Applications to educational technology. In: EdMedia: World Conference on Educational Media and Technology, Association for the Advancement of Computing in Education (AACE) (1999) 939–944
7. Juola, P.: Authorship attribution. Foundations and Trends in Information Retrieval **1**(3) (2008) 233–334
8. Jankowska, M., Milios, E., Kešelj, V.: Author verification using common n-gram profiles of text documents. In: Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, Dublin, Ireland, Dublin City University and Association for Computational Linguistics (August 2014) 387–397
9. Doyle, J.: Automatic evaluation of student essays using n-gram analysis techniques. Master's thesis, Dalhousie University (2007)
10. Stamatatos, E.: Author identification using imbalanced and limited training texts. In: Proceeding of the 18th International Workshop on Database and Expert Systems Applications, DEXA'07, Regensburg, Germany (September 2007) 237–241
11. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research **12** (2011) 2825–2830
12. Bird, S., Loper, E., Klein, E.: Natural Language Processing with Python. O'Reilly Media Inc. (2009)
13. Lemaître, G., Nogueira, F., Aridas, C.K.: Imbalanced-learn: A Python toolbox to tackle the curse of imbalanced datasets in machine learning. Journal of Machine Learning Research **18**(17) (2017) 1–5