Automated Essay Scoring Using the KNN Algorithm

Li Bin, Lu Jun, Yao Jian-Min, Zhu Qiao-Ming

Provincial Key Laboratory of Computer Information Processing Technology
Soochow University, Suzhou, China, 215006
libin0033@163.com, lujun_59@126.com, {jyao, qmzhu}@suda.edu.cn

***Abstract*—The K-Nearest Neighbor (KNN) algorithm for text categorization is applied to CET4 essays. In this paper, each essay is represented by the Vector Space Model (VSM). After removing the stop words, we chose the words, phrases and arguments as features of the essays, and the value of each vector is expressed by the term frequency and inversed document frequency (TF-IDF) weight. The TF and information fain (IG) methods are used to select features by predetermined thresholds. We calculated the similarity of essays with cosine in the KNN algorithm. Experiments on CET4 essays in the Chinese Learner English Corpus (CLEC) show accuracy above 76% is achieved.**

Keywords- automatic essay scoring; KNN algorithm; vector space model; feature selection

I. INTRODUCTION

While English writing is an essential part of the educational process, many raters find that scoring students' writing is one of the most expensive and time consuming activities for education assessment [1]. Related works include PEG, the earliest AES system developed by Page et al [2, 3], which scores based on shallow linguistic features, including essay length, number of prepositions, number of relational pronouns, variation of word length, etc. The IEA system [4], developed in the late 1990s based on latent semantic analysis, shows an approximate consistency of 85-91%. Burstein et al developed the E-Rater system [5] based on Microsoft parsing tools for the ETS in the TOEFL and GRE. For a test of 750,000 essays in GMAT, E-Rater has gained a consistency of 97% with the human scores [6]. BETSY is developed by Lawrence M. Runder [7], which gains a precision of 80% on 80 essays after training on 462 essays with a 2-category model.

This paper presents an approach to essay scoring that builds on the text categorization model which incorporates K-Nearest Neighbor (KNN) algorithm [8]. Transforming the essays into the vector space model (VSM), TF-IDF and IG are applied for feature selection from the feature pool of words, phrases and arguments. After training for the KNN algorithm, a precision over 76% is achieved on the CLEC corpus.

II. AUTOMATED ESSAY SCORING WITH KNN

A. Transform the Problem to Text Categorization

Several studies have reported favorably on computer grading of essays. The current systems have returned grades that correlated significantly with human raters. To implement our text categorization approach, we transform the desired scoring to a three-point categorical or nominal scale (e.g., A, B, C) and use a large set of features expressed by words, phrases and arguments.

B. Transform Essays into Vectors

Because the KNN algorithm we adopt is based on the attribute-value representation, we transform each essay into a vector in the model of vector space. First of all, essays need to be pre-processed. There are large numbers of common articles, pronouns, adjectives, adverbs, and prepositions in the English writings. Some text categorization studies have reported improved accuracy with trimmed stop words [9].

Then the transformation takes place. The vector space model can be represented as follow:

$$d_j = (w_{1j}, w_{2j}, w_{3j}, ..., w_{mj}) \quad (1)$$

d_j denotes the jth essay, and w_{ij} denotes the weight of the ith feature in jth essay, which represents the weight of the features.

The most commonly used weighting method is the TF-IDF term-weighting method [10]. Denote TFIDF(i,j) as the ith coordinate of the jth transformed essay.

$$TFIDF(i,j) = TF(i,j) \cdot \log \frac{N}{DF(i)} \quad (2)$$

TF(i,j) denotes the times of the ith feature in the jth essay. N is the number of all essays in the corpus, and DF(i) counts the essays containing the ith feature. Then, each essay of different lengths is transformed into a vector that has the same length in the vector space.

C. Feature Selection

The dimensionality of the vector space may be very high, which is disadvantageous in machine learning for complexity problem and over-learning. And also in order to get more effective features, dimension reduction techniques are necessary. Two methods are included in this study, which use the feature-goodness criteria to achieve a desired degree of feature elimination from the full features of an essay.

1) Term Frequency (TF)

We computed the feature frequency for each unique feature in the training corpus and removed from the vector space those features whose frequency is lower than the predetermined threshold. The basic assumption is that rare features are either non-informative for category prediction, or not influential on the global performance.

2) Information Gain (IG)

Information gain is frequently employed as a feature-goodness criterion in the field of machine learning [9, 11]. The

978-0-7695-3336-0/08 $25.00 © 2008 IEEE
DOI 10.1109/CSSE.2008.623

commonly used measure of information from information theory is entropy [12, 13]. Entropy is defined as:

$$E(S) = \sum_{j=1}^{J} -p_j \log_2 p_j \tag{3}$$

Here, p_j is the probability of belonging to the category j.

Entropy can be viewed as a measure of the uniformity of a distribution and has a maximum value when $p_j = 1/J$ for all j. The goal is to have a peaked distribution of p_j. The potential information gain then is the reduction in entropy.

We defined D as the set of the essays, F={f1,f2,…,fm} as the set of features, C={c1,c2,…,c|C|} as the set of categories. And the information gain G(D, fi) can be transformed as follows[14]:

$$\begin{aligned} G(D,f_i) = & -\sum_{c_k \in C} p(c_k)\log_2 p(c_k) \\ & + p(\overline{f_i}) \sum_{c_k \in C} p(c_k / \overline{f_i})[\log_2 p(c_k / \overline{f_i})] \\ & + p(f_i) \sum_{c_k \in C} p(c_k / f_i)[\log_2 p(c_k / f_i)] \end{aligned} \tag{4}$$

Here, p(ck) is the probability of the category k, $p(\overline{f_i})$ is the probability that feature i doesn't appear. $p(f_i)$ is the probability of feature i appears. The $p(c_k / \overline{f_i})$ indicates the probability that category k can be chosen when feature f_i doesn't appear, and $p(c_k / f_i)$ indicates the probability that category k can be chosen when feature f_i appears. The probability is calculated from the training corpus as:

$$p(c_k / \overline{f_i}) = C(c_k, \overline{f_i}) / C(D, \overline{f_i}) \tag{5}$$

$$p(c_k / f_i) = C(c_k, f_i) / C(D, f_i) \tag{6}$$

Here, $C(c_k / \overline{f_i})$ is the number of essays of the category k which doesn't contained the feature f_i in training corpus, $C(c_k, f_i)$ is the number of essays of the category k which contain the feature f_i in the training corpus, and the $C(D, \overline{f_i})$ is the number of the total essays which doesn't contain the feature fi, $C(D, f_i)$ is the number of the total essays which contain feature f_i.

D. The Model of Text Categorization

For KNN algorithm, all different features in the training essays compose a vector space. Each vector is expressed by the TF-IDF weights of the features in all training essays. For the test essays, the goal is to search the k-nearest neighbors from the training essays. So each test essay is also viewed as a special vector with the TF-IDF weight. We compute the similarity of the test essay with all of the training essays using cosine formula. The cosine formula is defined as follows [14]:

$$S(d_i, d_j) = \cos(\alpha) = \frac{\sum_{k=1}^{m} w_{ki} \cdot w_{kj}}{\sqrt{\sum_{k=1}^{m} w_{ki}^2} \cdot \sqrt{\sum_{k=1}^{m} w_{kj}^2}} \tag{7}$$

Here, all parameters are defined in as in formula (1). We sorted the results by decreasing order and select the first K essays. Then we classify the essay to the same category containing the most essays in those K essays.

III. EXPERIMENT DESIGN AND RESULTS

We used the CET4 essays of Chinese Learner English Corpus (CLEC) for this study. We choose the topic "Global shortage of Fresh Water" that contains 271 applicable essays for three score levels. Because there is just one essay whose point is below 5, we signed 0-10 points essays as "B"; 11-15 points essays as "A"; Twenty-five essays from each group were randomly selected to be used as the test samples. The remaining 221 essays were used as training.

A. Experimental Settings

Before feature selection, we remove the words that are in the stop word list [15]. Then the two feature selection methods are evaluated with a number of different feature-removal thresholds. At a high threshold, it is possible that all the features in an essay are below the threshold. To avoid removing all the features from an essay, we added a meta-rule to the process [15]. Considering that features which occur lots of times in every category offer little information, we remove those features by different thresholds. And here, the features have an occurrence of more than 40 times are removed.

B. Performance Measures

Categorization effectiveness is usually measured by precision and recall rate [14, 16]:

Precision: $$p = \frac{TP}{TP + FP} = \frac{\sum_{i=1}^{|C|} TP_i}{\sum_{i=1}^{|C|} (TP_i + FP_i)} \tag{8}$$

Recall: $$r = \frac{TP}{TP + FN} = \frac{\sum_{i=1}^{|C|} TP_i}{\sum_{i=1}^{|C|} (TP_i + FN_i)} \tag{9}$$

C. Primary Results

In our study, we find the results are better when K=3 or 5. Fig. 1 shows the average accuracy of words, phrases and arguments with different thresholds of TF when selecting features. From Fig. 1, we can see the best thresholds of TF are 20, 30 and 40 respectively when selecting words, phrases and arguments as features. Below these best thresholds of TF, The accuracy of different K tends to increase as TF increases. While over these best thresholds of TF, the accuracy tends to decrease as TF increases. The performance is mainly due to the features decrease when increasing TF over the best thresholds. We can see that the maximum value of average accuracy is 76% at the threshold of 40 when argument is selected as features.

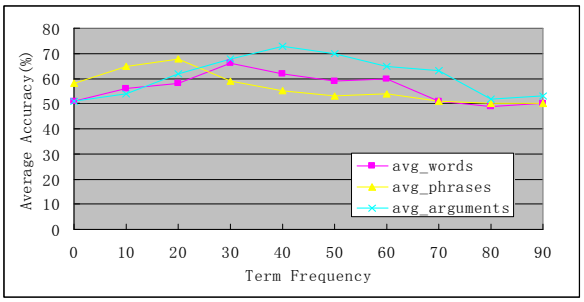

Figure 1. The Average Accuracy Based on TF

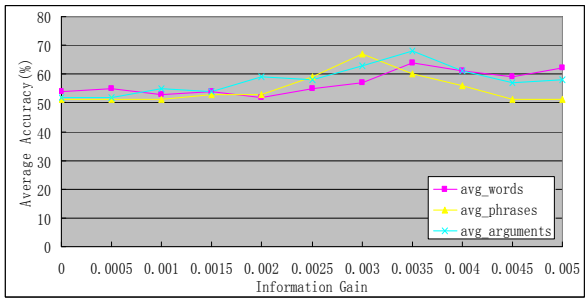

Figure 2. The Average Accuracy Based on IG

Fig. 2 shows the average accuracy of words, phrases and arguments with different thresholds of IG above 0. We can see the best thresholds of IG are 0.0025, 0.003 and 0.0035 respectively when the features are words, phrases and arguments. And the max value of average accuracy is 67% at the thresholds of 0.003 when phrase is selected as features.

Compared to TF, the maximum value of average accuracy of IG is lower.

Table 1 shows the result using different features when K=3 and K=5. The maximum accuracy is 76% when K=3, selecting phrases as feature with TF.

TABLE I. THE RESULT OF THE TESTING

TF	K=3			K=5			average		
	words	Phrases	arguments	words	phrases	arguments	avg_words	avg_phrases	avg_arguments
0	0.52	0.54	0.52	0.5	0.62	0.5	0.51	0.58	0.51
10	0.52	0.66	0.54	0.6	0.64	0.54	0.56	0.65	0.54
20	0.54	**0.7**	0.6	0.62	0.66	0.64	0.58	0.68	0.62
30	0.58	0.6	0.68	**0.74**	0.58	0.68	0.66	0.59	0.68
40	0.6	0.56	**0.76**	0.64	0.54	0.7	0.62	0.55	0.73
50	0.56	0.54	0.7	0.62	0.52	0.7	0.59	0.53	0.7
60	0.56	0.52	0.64	0.64	0.56	0.66	0.6	0.54	0.65
70	0.48	0.5	0.66	0.54	0.52	0.6	0.51	0.51	0.63
80	0.48	0.5	0.54	0.5	0.5	0.5	0.49	0.5	0.52
90	0.44	0.5	0.54	0.56	0.5	0.52	0.5	0.5	0.53

IG	K=3			K=5			average		
	words	Phrases	arguments	words	phrases	arguments	avg_words	avg_phrases	avg_arguments
0	0.56	0.5	0.5	0.52	0.52	0.54	0.54	0.51	0.52
0.0005	0.58	0.52	0.5	0.52	0.5	0.54	0.55	0.51	0.52
0.001	0.54	0.5	0.52	0.52	0.52	0.58	0.53	0.51	0.55
0.0015	0.54	0.52	0.52	0.54	0.54	0.56	0.54	0.53	0.54
0.002	0.52	0.54	0.54	0.52	0.52	0.64	0.52	0.53	0.59
0.0025	0.56	0.6	0.56	0.54	0.58	0.6	0.55	0.59	0.58
0.003	0.58	**0.68**	0.58	0.56	0.66	**0.68**	0.57	0.67	0.63
0.0035	0.62	0.66	**0.68**	**0.66**	0.54	**0.68**	0.64	0.6	0.68
0.004	0.62	0.56	0.6	0.6	0.56	0.62	0.61	0.56	0.61
0.0045	0.6	0.5	0.58	0.58	0.52	0.56	0.59	0.51	0.57
0.005	0.62	0.5	0.6	0.62	0.52	0.56	0.62	0.51	0.58

IV. Conclusion and Future Work

We presented the KNN algorithm for automatic essay scoring based on the well developed text categorization modeling. Based on features of words, phrases and arguments, we divide the essays into two categories. The result is quite promising. With the different methods of feature selection, we are able to achieve 76% accuracy. In our study, the result is much better when K=3 than K=5, and arguments tend to outperform words and phrases. Removing the features with high frequency in each category tends to improve accuracy. Because the features are very similar of the essays about the same item, the accuracy is a little lower than KNN algorithm used in general text categorization.

In next phase of our research, we will do our best to take measures to get more features (e.g.; the construction of sentences) with different methods to improve our accuracy. As the initial corpus of scored essays is very small, we will try to get larger corpus. And we will consider more score categories, and more detailed categories.

[1] Blok H, de Glopper K. Large scale writing assessment. In: L. Verhoeven, J. H. A. L. De Jong (Eds.).The construct of language proficiency: Applications of psychological models to language assessment. Amsterdam, Netherlands: John Benjamins Publishing Company, 1992. 101-111.

[2] Burstein J, Leacock C, Swartz R. Automated evaluation of essays and short answers. In: M. Danson (Ed.). Proceedings of the Sixth International Computer Assisted Assessment Conference. Loughborough University, Loughborough, UK, 2001.

[3] Foltz P W, Laham D. The Intelligent Essay Assessor: Application to Education Technology. In: T.K. Landauer (Ed.). Interactive Multimedia Electronic Journal of Computer-Enhanced Learning. Wake Forest University, 2000.

[4] Mitchell T, Russel T, Broomhead P, Aldridge N. Towards robust computerized marking of free-text responses. In: M. Danson (Ed.). Proceedings of the Sixth International Computer Assisted Assessment Conference. Loughboroug University, Loughborouh, UK, 2002.

[5] Page, E.B. Grading essays by computer: Why the controversy? Handout for NCME Invited Symposium.1996.

[6] Rudner L M Liang T. Automated essay scoring using Bayes' Theorem. The Journal of Technology, Learning and Assessment, 2002, 1(2): 3-21.

[7] Valenti S, Cucchiarelli A, Panti M. Computer based assessment systems evaluation via the ISO9126 quality model. Journal of Information Technology Education, 2002, 1 (3), 157-175.

[8] Yang Y, Liu X. A re-examination of text categorization methods. In: Proceedings 22nd Annual International ACM Sl2 GIR Conference on Research and Development in Information Retrieval. Berkeley: ACM Press, 1999. 42 - 49.

[9] Mitchell T. Machine Learning. Annual Review of Computer Science, 1990, 4(1), 417-433.

[10] A Aizawa. An information-theoretic perspective of tf-idf measures. Information Processing and Management, 2003, 39(1), 45-65.

[11] J.R. Quinlan. Induction of decision trees. Machine Learning, 1986, 1(1):81-106.

[12] Cover T M, Thomas J A. Elements of information theory. New York: Wiley, 1991.

[13] Yang Y, Pedersen J P. A comparative study on feature selection in text categorization. Proceedings of the Fourteenth International Conference on Machine Learning. San Francisco: Morgan Kaufmann Publishers, 1997, 412 - 420.

[14] Fabrizio, Sebastiani. Machine learning in automated text categorization. ACM Computing Surveys (CSUR), 2002, 34(1), 1-47.

[15] G Salton. Automatic Text Processing: the Transformation, analysis, and retrieval of information by computer. Boston: Addison-Wesley Longman Publishing Co., 1989.