DOI: 10.21917/ijsc.2014.0102

EXPERIMENTS TOWARDS DETERMINING BEST TRAINING SAMPLE SIZE FOR AUTOMATED EVALUATION OF DESCRIPTIVE ANSWERS THROUGH SEQUENTIAL MINIMAL OPTIMIZATION

Sunil Kumar C[1] and R. J. Rama Sree[2]
[1]*Research and Development Center, Bharathiar University, India*
E-mail: sunil_sixsigma@yahoo.com
[2]*Department of Computer Science, Rashtriya Sanskrit Vidyapeetha, India*
E-mail: rjramasree@yahoo.com

Abstract
With number of students growing each year there is a strong need to automate systems capable of evaluating descriptive answers. Unfortunately, there aren't many systems capable of performing this task. In this paper, we use a machine learning tool called LightSIDE to accomplish auto evaluation and scoring of descriptive answers. Our experiments are designed to cater to our primary goal of identifying the optimum training sample size so as to get optimum auto scoring. Besides the technical overview and the experiments design, the paper also covers challenges, benefits of the system. We also discussed interdisciplinary areas for future research on this topic.

Keywords:
Descriptive Answers, Auto Evaluation, LightSIDE, Machine Learning, SVM, Sequential Minimal Optimization

1. INTRODUCTION

Delays in evaluation of examination answer scripts by examiners are a problem highlighted often by media [14]. One of the reasons for the delay is non-availability of evaluators or availability of very few qualified evaluators. With millions of students giving various academic exams every year this problem is going to be a very challenging one. Overloading examiners with more number of answer scripts to evaluate may lead to issues with quality during evaluation. It is astonishing to find some reports from newspapers on how marks were increased or decreased when students apply for revaluation of their answer scripts [15],[16].

A number of software packages evolved in the area of online examination automation however all of the currently available packages provide support only for auto evaluation of objective type answers for questions of type true / false, multiple choice etc.

In the current education system, it is proved through numerous researches that objective type evaluation of an individual is just not enough and that evaluation thru descriptive questions of type essays, short answers is very much required [17]. This requirement brings into picture the need for online examination systems to provide support for auto-evaluation of subjective answers provided by examinees.

Evaluation of answers and providing a scoring is a classification task where in the human evaluator or the system is supposed to interpret the answer and classify the answer into one of the possible rubrics pre-allocated for the answer. A human evaluator is capable of evaluating and classifying the answer because of his experience and the reference material he has got. Similarly, we believe supervised learning method can be applied to classify the answers into appropriate rubric based on the likelihood suggested by training samples.

A simple approach towards problem solving is to leverage data mining techniques where in words are extracted from the answers then compare the same with words that were previously extracted from training samples to obtain the score. Unfortunately, this method is widely criticized because the method simply does the word matching rather than interpreting the actual concept of the document. Our literature review suggested that some research was already done in this area and some suggestions to overcome the problems are as below –

- Ontology enhanced representation. That is, using ontology to capture the concepts in the documents and integrate the domain knowledge of individual words into the terms for representation. For instance, Hotho et al. developed different types of methods to compile the background knowledge embodied in ontologies into text documents representation and improved the performance of document clustering [1]. Such kind of works also can be found in [2, 3].
- Linguistic unit enhanced representation. This method makes use of lexical and syntactic rules of phrases to extract the terminologies, noun phrases and entities from documents and enrich the representation using these linguistic units. For instance, Lewis compared the phrase-based indexing and word-based indexing for representation for the task of document categorization [4]. His result showed that the phrase indexing cannot improve the categorization in most cases because of the low frequencies of most phrases. Such kind of work can also be found in [5] which used multi-words to improve the effectiveness of text-retrieval system.
- Word sequence enhanced representation. This method ignores the semantics in documents and treats the words as string sequences. Text representation using this method is either on words' group based on co-occurrence or a word sequence extracted from documents by traditional string matching method. In this aspect, Li used the generalized suffix tree to extract the frequent word sequences from documents and used the frequent word sequences for text representation to propose the CFWS clustering algorithm [6]. Similar work can be founding [7–9]. Particularly, the N-gram-based representation [10] can also be categorized as this type for it also ignores the semantics and meaning of individual words.

Our experiments discussed later in this paper use a combination of the methods from Linguistic unit enhanced representation and Word sequence enhanced representation. The primary focus of this paper is to evaluate the interdependency between samples and samples size of the training set and the results obtained during classification. The scope of this paper is not to evaluate the interdependency between classification algorithm and the classification results.

The rest of this paper is organized as follows. Section 2 discusses experimental setup and the preliminaries of the tools and techniques used in this paper along with the related work. Section 3 describes the results obtained from the experiments and the conclusion remarks.

2. EXPERIMENTAL SETUP

The setup in which the experiments are conducted for this paper are specified and the related work of each topic is introduced.

2.1 DATA COLLECTION AND DATA CHARACTERISTICS OF TRAINING DATA

In February 2012, The William and Flora Hewlett Foundation (Hewlett) sponsored the Automated Student Assessment Prize (ASAP) [18] to machine learning specialists and data scientists to develop an automated scoring algorithm for student-written essays. As part of this competition, the competitors are provided with hand scored essays under 8 different prompts. 5 of the 8 essays prompts are used for the purpose of this research.

All the graded essays from ASAP are according to specific data characteristics. All responses were written by students ranging in grade levels from Grade 7 to Grade 10. On average, each essay is approximately 150 to 550 words in length. Some are more dependent upon source materials than others. Each question for which answers are written, the number of answers provided for each question varies from one another. For example, the lowest amount of training data is 1,190 answers, randomly selected from a total of 1,982. Each answer in the training data set is provided with a score by two human evaluators. There may be certain cases where the scores provided by independent evaluators are different; this is due to the fact that sometimes human evaluators also do not agree on providing same score for an answer [19]. Even the test data we used for our experiments come with two scores provided by human evaluators. Our aim is to predict and comply with one of the human scores given the situation of multiple score exists.

2.2 LightSIDE PLATFORM

For the purpose of designing and evaluating our experiments, we have used a machine learning interface called LightSIDE.

LightSIDE (Light Summarization Integrated Development Environment) is a free and open source offering from Carnegie Mellon University (TELEDIA lab).This program has a user-friendly interface and it incorporates numerous options to develop and evaluate machine learning models. These models can be utilized for a variety of purposes, including automated essay scoring. LightSIDE focuses on the syntactical elements of the text rather than semantics [20].

LightSIDE cannot evaluate any random content or creative content. The automated evaluation we are referring to is for a specific context. LightSIDE can be trained with answers on specific questions and later automated assessment is relevant only for those answers written for specific questions that the earlier training data set belongs to.

Using LightSIDE to achieve AES involves 4 different steps [18] as shown below,

a. Data collection and date input file formatting - LightSIDE Labs recommends at least 500 data set items for each question that the system is getting trained on. [15] Once the training data set is available, Data should be contained in a .csv file, with every row representing a training example, except the first, which lists the names of the fields of the data. At least one column in the data should be the label and the other columns can be text and meta-data related to the training example. Light SIDE's GUI interface provides the user with an option to load the input file.
b. Feature extraction –From the input training data set file, user can specify on the LightSIDE GUI the features to be extracted for the purpose of creating a feature table which can later be used to create machine learning model.
c. Model building - With the feature table in hand, one can now train a model that can replicate human labels by selecting the desired machine learning algorithm from LightSIDE's GUI interface and also the GUI can be used to set the various parameters applicable. Models's performance can also be tested with default 10 fold cross validation or other validation options available on LightSIDE GUI.
d. Predictions on new data – Using the model that is built, new data can be loaded and the classification auto essay scoring task can be carried so as to get the resultant predications on the new data. New data presented for evaluation by LightSIDE also need to abide the input formatting rules as mentioned in step an above.

2.3 STATISTICAL FEATURE EXTRACTION

Though LightSIDE offers capabilities to extract advanced features from training data set, we have limited our self to basic text features for the purpose of this experiment. Below features are extracted from input training data set to build feature table –

a. Unigrams - An n-gram of size 1 is referred to as a "unigram".
b. Bigrams - An n-gram of size 2 is a "bigram" (or, less commonly, a "diagram").
c. Trigrams - An n-gram of size 3 is a "trigram".
d. POS Bigram – Part of Speech Bigrams.
e. Line Length.
f. Remove Stop words - Stop words are the short functional words such as the, is, at, which, and on etc., these do not add any value from a meaning perspective to the sentence however syntactically are a must for the sentence.
g. Stem N-grams – Stemming is a process of reducing a word into its root or base form. For example the root form of the words experimentation, experimental is experiment.

2.4 SEQUENTIAL MINIMAL OPTIMIZATION (SMO)

Previous work undertaken on auto essay scoring using LightSIDE suggested that SMO consistently performed better than other machine learning algorithms [12] available through LightSIDE. We used the SMO (Regression) for our research purposes.

SMO by itself is not a classification method. However, SMO can be considered as a part of a classification method called Support Vector Machine [13].

2.5 TEST DATA

In each of the 5 training data sets used for our research, we used only 1500 data items for training purposes. For each data set, we separated a set of 142 samples and another set of 25 samples to use as test data sets.

We ensured that the test data sets are non-intersecting with training data sets i.e., none of the test samples are used as part of training data sets.

2.6 MODEL BUILDING AND HYPOTHESIS

Using each data set, we built models using 500, 600, 700, 800, 900, 1000, 1100, 1200, 1300, 1400, 1500 training data items.

Our hypothesis is that the percentage of correct score prediction betters with increase of number of training data items. For example if the model built with 500 training samples yields X% correct predictions then the correct predictions percentage using model built with 600 samples is always greater than X%.

2.7 MEASUREMENT OF PREDICTIONS

We observed that our models were predicting scores in decimals whereas the original data set only had whole number rubrics. In certain cases we observed that negative scores were predicted to some test samples. From our dataset, we observed that this is not a possibility as all scores start with 0 and move upwards. Although there were only few cases, we observed that the predicted score was more than the upper boundary rubric possible.

Before analysing our data, as a contingent measure, we rounded all decimal predicted scores to nearest whole number. We replaced all negative predicted scored with the lowest possible score of 0. All predicted scores which were more than the upper boundary of possible scores; we replaced them with highest possible score.

We then compared the obtained predicted scores with that of the manual scores provided by human evaluators. We considered the predicted score to be correctly predicted if it complies with at least one of the two scores provided by human evaluators. For each prompt, we calculated the percentage of test samples correctly predicted separately for the 25 test data set samples and 142 test data samples.

Once all calculations are over, we averaged the percentages by training sample size. This is to identify the best training data set size that yields the highest percentage of correct predicted scores. As per our hypothesis, we expected that to be 1500 samples training data set!

3. RESULTS AND CONCLUSION

Table.1. Percentage of correctly predictions on 25 test samples

	% correctly predicted score					
Training sample size	Data Set1	Data Set2	Data Set3	Data Set4	Data Set5	Average scores
500	44	44	68	84	72	62.4
600	44	56	68	72	76	63.2
700	56	60	84	68	76	68.8
800	56	64	88	76	64	69.6
900	60	72	80	84	64	72
1000	64	56	76	84	68	69.6
1100	52	60	68	80	76	67.2
1200	52	48	76	76	72	64.8
1300	48	60	76	76	68	65.6
1400	48	44	80	80	72	64.8
1500	60	48	68	84	76	67.2

Table.2. Percentage of correctly predictions on 142 test samples

	% correctly predicted score					
Training sample size	Data Set1	Data Set2	Data Set3	Data Set4	Data Set5	Average scores
500	51.4	62.67	68.3	85.21	85.21	70.5
600	53.52	59.15	62.67	85.91	80.98	68.4
700	54.22	52.81	69.71	81.69	85.21	68.7
800	50	51.4	69.01	80.28	83.09	66.7
900	51.4	47.88	69.01	81.69	80.98	66.1
1000	48.59	48.59	70.42	87.32	84.5	67.8
1100	58.45	50	73.94	85.91	84.5	70.5
1200	57.04	49.29	70.42	87.32	85.21	69.8
1300	55.63	52.11	70.42	84.5	85.91	69.7
1400	54.92	52.81	72.53	80.98	85.21	69.2
1500	54.22	54.92	71.12	88.73	83.09	70.4

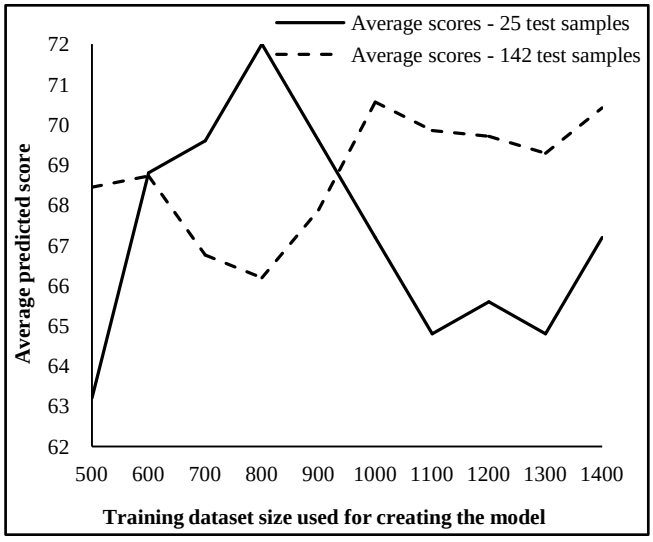

Fig.1. Line chart of average predicted scores %

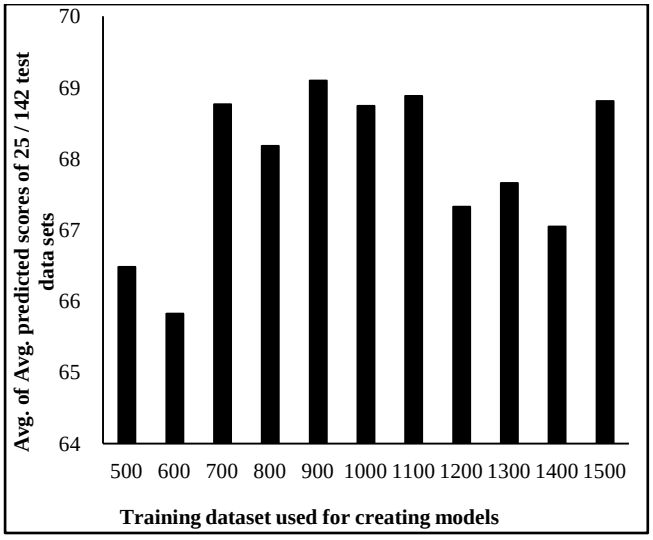

Fig.2. Bar chart of average of averages of predicted scores %

2.7 RESULTS & DISCUSSION

The results presented did not conclude any clear pattern in terms of the relationship between training samples used to build the models and the predicted scores. It is interesting to observe that the accuracy of prediction differed with each model and with each test dataset. The predication score compliance percentage ranged from 44% to a whopping 88.73%!

From the results obtained it is very evident that our hypothesis that "the correctly predicted scores will better align with human scorers due to increase of training samples" is incorrect.

In terms of determining the best sample size, we see that average percentages vary only very minimally. The range of averages is laid between 65.823 and 69.096. If a deliberate optimal option needs to be chosen then from the results we see that at 900 training samples the average prediction score seems to be touching the highest with 69.096%.

2.7 LIMITATIONS AND FUTURE DIRECTIONS

While we are not able so far identify the clear reason for this broad range of correct prediction percentages, one speculation is that the training data set itself i.e., quality of the training data and the features or characteristics of the training data. Further research is required to identify the root cause of higher compliance percentage only in some cases. Yet another perspective to be studied is about how well the test data features aligned with training data features in cases where we obtained high compliance percentage. This study will reveal the relationships between test data and training data features.

In our future research, we would also like to study the behavior of models when the test data set is merely a subset of training data set.

Conducting similar kind of experiments and comparison with other machine learning algorithms such as J48, Naïve Bayes is another direction to work on.

Our current research focused on basic features extraction from training data in order to build models. We would like to extend our research on the same topic by including additional features such as Punctuations, Binary N-Grams, Differentiating text columns etc., to build models for scores prediction.

REFERENCES

[1] A. Hotho, S. Staab and G. Stumme, "Ontologies Improve Text Document Clustering", *Proceedings of the 3rd IEEE International Conference on Data Mining*, pp. 541 – 544, 2003.

[2] S. Scott and S. Matwin, "Text classification using WordNet Hypernyms", *Proceedings of the COLING/ACL Workshop on Usage of WordNet in Natural Language Processing Systems*, pp. 45 – 52, 1998.

[3] M. B. Rodriguez, J. M. G. Hidalgo and B. D. Agudo, "Using WordNet to complement training information in text categorization", *Proceedings of 2nd International Conference on Recent Advances in Natural Language Processing*, pp. 353 – 364, 1997.

[4] D. D. Lewis, "An evaluation of phrasal and clustered representation on a text categorization task", *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 37 – 50, 1992.

[5] R. Papka and J. Allan, "Document classification using multiword features", *Proceedings of the Seventh International Conference on Information and Knowledge Management Table of Contents*, pp. 124 – 131, 1998.

[6] Y. J. Li, S. M. Chung and J. D. Holt, "Text document clustering based on frequent word meaning sequences", *Data & Knowledge Engineering*, Vol. 64, No. 1, pp. 381 – 404, 2008.

[7] B. C. M. Fung, K. Wang and M. Ester, "Hierarchical document clustering using frequent item sets", *Proceedings of SIAM International Conference on Data Mining*, pp. 59– 70, 2003.

[8] T. B. Ho and K. Funakoshi, "Information retrieval using rough sets", *Journal of the Japanese Society for Artificial Intelligence*, Vol. 13, No. 3, 424 – 433, 1998.

[9] T. B. Ho and N. B. Nguyen, "Non-hierarchical document clustering based on a tolerance rough set model", *International Journal of Intelligent Systems*, Vol. 17, No. 2, pp. 199 – 212, 2002.

[10] W. B. Cavnar and J. M. Trenkle, "N-Gram based text categorization", *Proceedings of 3rd Annual Symposium on Document Analysis and Information Retrieval*, pp. 161 – 169, 1994.

[11] Elijah Mayfield, David Adamson and Carolyn Penstein Rose, "*LightSIDE Researcher's user manual*", pp. 5 – 9, 2013.

[12] Syed M. Fahad Latifi, Q. Guo, M. J. Gierl, A. Mousavi and K. Fung, "Towards Automated Scoring using Open-Source Technologies" *Annual Meeting of the Canadian Society for the Study of Education Victoria*, pp. 13 – 14, 2013.

[13] Platt John, "Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines", Technical Report, Microsoft Research, 1998.

[14] Staff Reporter, The Hindu, Bangalore, "Protest over delay in evaluation work",

http://www.thehindu.com/news/cities/bangalore/protest-over-delay-in-evaluation-work/article4214480.ece.

[15] The Times of India, Nagpur, "80 out of 83 score more after revaluation", http://articles.timesofindia.indiatimes.com/2011-07-15/nagpur/29777272_1_revaluation-results-rechecking-redressal-system.

[16] Sridhar Vivan, Bangalore Mirror, Bangalore, "Revaluation fails 100 'passed' PU students",
http://www.bangaloremirror.com/index.aspx?page=article&sectid=10&contentid=20110628201106282358189681f9dbf8.

[17] Siddhartha Ghosh, "e-Examiner: A System for Online Evaluation & Grading of Essay Questions", http://elearn.cdac.in/eSikshak/eleltechIndia05/PDF/05-e-Examiner%20A%20system%20for%20online%20evaluation%20&%20grading%20of%20essay%20questions-Sidharth-05.pdf.

[18] http://www.kaggle.com/c/asap-aes.

[19] Code for evaluation metric and benchmarks, https://www.kaggle.com/c/asap-aes/data?Training_Materials.zip, Accessed on 10 February 2012.

[20] http://lightsidelabs.com/our-technology.