International Journal of Computational Intelligence and Informatics, Vol. 3: No. 1, April - June 2013

Analysis of Effects of Feature Selection to ImproveModels Performance during Automated Evaluation of Descriptive Answers through Sequential Minimal Optimization

C Sunil Kumar
Research and Development Center,
Bharathiar University, Coimbatore, India
sunil_sixsigma@yahoo.com

R J Rama Sree
Rashtriya Sanskrit Vidyapeetha,
Tirupati, India
rjramasree@yahoo.com

***Abstract*- In this paper, we applied feature selection as a technique to eliminate the problem of huge number of features in text classification. We quantitatively analyzed the usefulness of various syntactical and semantic features of the text with regards to models' performance. Based on the results we derived general principles to apply during auto evaluation of descriptive answers.**

Keywords-Descriptive answers, auto evaluation, LightSIDE, Machine Learning, SVM, SMO, Sequential Minimal Optimization, Feature selection

I. INTRODUCTION

Evaluation of answers and providing a scoring is a hard classification task (assigning a single category to each document) 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. We believe supervised learning method can be applied to classify the answers into appropriate rubric based on the likelihood suggested by training samples. The supervised learning process requires extracting various text features from the documents meant as training set and then train using a sophisticated machine learning algorithm. One particular problem with text classification task is that depending on the document size, the number of features can be very large sometimes spanning into thousands too! The huge number of features is a major problem for training algorithm to perform effective learning and execution. Feature selection helps with sorting this problem of huge number of features. There are two ways available to reduce the number of features, the first one is Feature selection which eliminates the un-required features from the complete set of features – this means only some of the key features well contributing to the models performance are chosen and used [1].

The other approach is Feature transformation which computes new features that are functions of the old features i.e., the reduced new features somehow inherently represent the old features [2]. Techniques such as principal component analysis do the task of identifying patterns with in high dimension data and then compressing i.e. by reducing the number of dimensions, without much loss of information [3,4].For the scope of this research paper, we focused our research on feature selection only. The context is that there are multiple types of features that can be extracted from the text but for a person building models to perform the task of hard classification of text, it is a challenge to select the best type features that helps build an effective model. While each model can be treated specially in order to do feature selection so as to make it a better model our research goal in this paper is to probe and derive the general principles that apply to any model for feature selection. The rest of this paper is organized as follows. Section 2 discusses the data used, experimental setup, the preliminaries of the tools and techniques used in this paper along with the related work. Section 3 describes the models built and measurements made during the experiments. Finally, analysis of results, concluding remarks and further research plans are indicated in Section 4.

II. EXPERIMENTAL SETUP

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

A. 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) 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 50 words in length.

Some are more dependent upon source materials than others. The number of training essays for each prompt (question) vary. For example, the lowest amount of training data is 1,190 essays, randomly selected from a total of 1,982. The data contains ASCII formatted text for each essay followed by one or more human scores, and (where necessary) a final resolved human score. Where it is relevant, more than one human score exists, so as to signify the reliability of the human scorers.

For the purpose of evaluation of the performance of the model, we considered the score predicted by the model to comply with one of the human scores given the situation of multiple scores. The data used for training, validation and testing the models are answers written by students for 5 different questions. Data for a question is considered as one unique dataset. So, we have a total of 5 datasets. The questions that students are asked to provide responses to are from Chemistry, English Language Arts and Biology.

B. 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.

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–

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. 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 thedata. 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.

C. 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 "digram").
c) Trigrams - An n-gram of size 3 is a "trigram".
d) POS Bigram – Part of Speech Bigrams. Traditional grammar classifies words based on eight parts of speech: the verb, the noun, the pronoun, the adjective, the adverb, the preposition, the conjunction, and the interjection however LightSIDE's parts of speech are based in computational linguistics research with more than 30 possibilities such as "VBP" (a non-third-person singular verb in the present tense) or "PRP" (a personal pronoun, such as "he" or "we"). There are also some specialized tags like "BOL," which simply represents the start of a paragraph, and "EOL," which is the same for the end of a paragraph.
e) Word/POS Pairs - Word / POS pairs is a decorator for unigram extraction that adds part-of-speech information to extracted unigram features.
f) Stop words - The most common, short function words, such as the, is, at, which, and on.
g) Stemming – It is a process of reducing inflected (or sometimes derived) words to their stem, base or root form—generally a written word form.
h) Punctuations - unigrams representing things like periods, commas, or quotation marks

D. Sequential Minimal Optimization (SMO)

Previous work undertaken on auto essay scoring using LightSIDE suggested that SMO consistently performed better than other machine learning algorithms [5] 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 [6].

E. Training Data, Test Data Size

In each of the 5 training data sets used for our research, the training set is 900 samples in size. Our previous research for determining appropriate sample size for automated essay scoring using SMO revealed that using 900 samples for training proved to yield slightly better results than using other sample sizes therefore the decision to use 900 samples as the training sample size.

For each data set, we separated a set of 100 samples to use as test data set. 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.

F. *Measurement of Predictions*

We observed that our models were predicting scores in decimals whereas the original data set (human provided scores) 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. We also had a challenge in terms of considering the values after the decimal point in the predicted scores.

We were not sure if any value post the decimal point in the predicted score needs to be rounded or if the score need to be ceiled else if the predicted score needs to be used floored. A separate analysis is performed to confirm the accurate action. Based on the analysis, it was confirmed that the predicted scores accuracy percentage on test samples was more when the scores where rounded therefore we rounded all decimal predicted scores. We also 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. Once all calculations are over, we observed for effects on predicted score percentage of the model if a particular set of features were excluded from the features used for training the model. The idea is to identify the generic set of features that can be excluded from features set used for training a model to perform automatic essay scoring but at the same time to ensure that the prediction accuracy is increased through elimination of noisy features.

III. MODELS BUILT AND MEASUREMENTS

Various models built during the experiments, the measurements obtained and various conclusions made through analysis of the measurements done during the experiments are described in this section. When models are built on LightSIDE, we used randomized 10-fold cross-validation in order to testing performance the models. Models' reliability is captured is reported through Pearson's Correlation Coefficient (R) and Mean Squared Error (MSE). Better models will have correlations closer to 1, and MSEs closer to zero.

A. *All Features Included models*

For each of the 5 data sets, we built a model by extracting unigrams, bigrams, POS Bigrams, Word / POS pairs, Punctuations, Stop words and with no Stemming performed on the dataset. The idea is to use the measurements obtained from the models as benchmark for comparison with other models built as part of these experiments considered for this paper. Below are the measurements obtained from the models built –

	Dataset 1	**Dataset 2**	**Dataset 3**	**Dataset 4**	**Dataset 5**
Number of features	38980	34179	42381	31693	29628
Correlation	0.795	0.152	0.672	0.77	0.813
Mean squared error	0.395	0.456	0.205	0.15	0.157
Percentage of prediction accuracy when predicted scores rounded	58	64	76	89	65
Percentage of prediction accuracy when predicted scores floored	55	44	64	86	72
Percentage of prediction accuracy when predicted scores ceiled	42	55	51	49	28

B. Models with Punctuations excluded

For each of the 5 data sets, we built a model by extracting unigrams, bigrams, POS Bigrams, Word / POS pairs, Stop words and with no Stemming performed on the dataset. Below are the measurements obtained from the models built.

	Dataset 1	**Dataset 2**	**Dataset 3**	**Dataset 4**	**Dataset 5**
Number of features	28493	25993	33672	25318	23500
Correlation	0.773	0.136	0.686	0.778	0.813
Mean squared error	0.432	0.458	0.199	0.146	0.157
Percentage of prediction accuracy when predicted scores rounded	58	68	79	91	64
Percentage of prediction accuracy when predicted scores floored	51	38	67	87	72
Percentage of prediction accuracy when predicted scores ceiled	45	60	52	44	27

C. Models with POS Bigrams excluded

For each of the 5 data sets, we built a model by extracting unigrams, bigrams, Punctuations, Word / POS pairs, Stop words and with no Stemming performed on the dataset. Below are the measurements obtained from the models built.

	Dataset 1	**Dataset 2**	**Dataset 3**	**Dataset 4**	**Dataset 5**
Number of features	38123	33350	41509	30988	28957
Correlation	0.794	0.171	0.671	0.782	0.824
Mean squared error	0.396	0.453	0.206	0.143	0.148
Percentage of prediction accuracy when predicted scores rounded	64	67	77	91	65
Percentage of prediction accuracy when predicted scores floored	51	40	65	87	70
Percentage of prediction accuracy when predicted scores ceiled	49	60	50	41	32

D. Models With Stop Words Removed

For each of the 5 data sets, we built a model by extracting unigrams, bigrams, POS Bigrams, Word / POS pairs, Punctuations and with no Stemming performed on the dataset. Below are the measurements obtained from the models built.

	Dataset 1	Dataset 2	Dataset 3	Dataset 4	Dataset 5
Number of features	33699	30604	38064	29389	27078
Correlation	0.787	0.146	0.685	0.783	0.819
Mean squared error	0.409	0.457	0.199	0.143	0.152
Percentage of prediction accuracy when predicted scores rounded	58	67	78	92	63
Percentage of prediction accuracy when predicted scores floored	53	42	65	86	72
Percentage of prediction accuracy when predicted scores ceiled	45	56	52	48	26

E. Models With Stemming Included

For each of the 5 data sets, we built a model by extracting unigrams, bigrams, POS Bigrams, Word / POS pairs, Stop words, Punctuations and with Stemming performed on the dataset

	Dataset 1	Dataset 2	Dataset 3	Dataset 4	Dataset 5
Number of features	36614	31562	39749	29521	27874
Correlation	0.795	0.148	0.678	0.778	0.809
Mean squared error	0.394	0.456	0.202	0.146	0.16
Percentage of prediction accuracy when predicted scores rounded	63	65	75	89	65
Percentage of prediction accuracy when predicted scores floored	54	41	66	87	70
Percentage of prediction accuracy when predicted scores ceiled	46	61	50	50	29

F. Models With Word/Pos Pairs Excluded

For each of the 5 data sets, we built a model by extracting unigrams, bigrams, POS Bigrams, Punctuations, Stop words and with no Stemming performed on the dataset. Below are the measurements obtained from the models built.

	Dataset 1	Dataset 2	Dataset 3	Dataset 4	Dataset 5
Number of features	36865	32260	39719	29304	27185
Correlation	0.794	0.148	0.672	0.765	0.812
Mean squared error	0.395	0.456	0.206	0.153	0.157
Percentage of prediction accuracy when predicted scores rounded	59	65	77	90	65
Percentage of prediction accuracy when predicted scores floored	55	43	64	86	72
Percentage of prediction accuracy when predicted scores ceiled	42	55	52	46	28

IV. RESULTS & CONCLUSIONS

A. Trials to accept rounded predicted score or floored predicted score or ceiled predicted score

Against each model built and measurement obtained, we identified the percentage accuracy score that is the highest from the rounded predicted scores, floored predicted score and ceiled predicted scores. For clarity purposes, we marked the highest score out of the three with black background. We observe that the rounded predicted accuracy percentage in 4 out of the 5 models is highest aligned with human scores therefore we proceed to accept rounded score as our general principle for measurements for further analysis in this paper.

B. Comparison of rounded predicted scores from various models to evaluate model's effectiveness

We compared rounded predicted scores of each model with that of benchmark values of the corresponding datasets obtained from all features included models. We have highlighted the scores that did not show any improvement with black background. The resultant measurement table is as below

Model Type - Rounded Prediction Accuracy %	Dataset 1	Dataset 2	Dataset 3	Dataset 4	Dataset 5
All Features Included Models	58	64	76	89	65
Models with Punctuations excluded	58	68	79	91	64
Models with POS Bigrams excluded	64	67	77	91	65
Models with stop words removed	58	67	78	92	63
Models with Stemming included	63	65	75	89	65
Models with Word/POS pairs excluded	59	65	77	90	65

Further, we measured the number of models that did not show improvement under each model type and called this measurement as feature score. The usefulness of a model type is computed in percentage (i.e., 100-(feature score/5)*100). The resultant measurement table is as below

Model Type - Rounded Prediction Accuracy %	Feature score	% of model usefulness
Models with Punctuations excluded	2	60
Models with POS Bigrams excluded	1	80
Models with stop words removed	2	60
Models with Stemming included	3	40
Models with Word/POS pairs excluded	1	80

It is clear from the data above that excluding Word / POS pairs or POS bigrams from the models will yield better prediction scores in 80% of the models. Similarly all other model efficiencies can be concluded from the table shown above. As we saw that eliminating features or including some useful features is resulting in better results, another curious question we had was how would a model behave when we exclude all no useful features and include only useful features. We called such a model as "Total noise reduction" model.

	Dataset 1	Dataset 2	Dataset 3	Dataset 4	Dataset 5
Number of features	25190	22847	29475	20915	19599
Correlation	0.762	0.145	0.675	0.804	0.828
Mean squared error	0.449	0.457	0.204	0.13	0.145
Percentage of prediction accuracy when predicted scores rounded	59	72	84	90	66

To build "Total noise reduction" model, we

- Included features – Unigrams, Bigrams, Trigrams, and Stemming.
- Excluded features – POS Bigrams, Stop words, Punctuations, Word / POS pairs.

Below are the measurements obtained and comparison with benchmark. The feature score and % of usefulness of the model are as below

Model Type - Rounded Prediction Accuracy %	Feature score	% of model usefulness
Models with Punctuations excluded	0	100

Clearly, the models built with bag of words features and with stemming implemented overtook models which included semantic features of sentence construction etc. Even when we compared the Total noise reduction models prediction accuracy with prediction accuracies obtained from other feature type models described in the previous sections, we see that the Total noise reduction models prediction accuracy trumped over the others. In a very few cases, we found the other models prediction accuracy trumped over total noise reduction models prediction accuracy however the difference in prediction accuracy was just 1 or 2 % over that of corresponding total noise reduction models' prediction accuracy. Therefore, the safe conclusion that can be made for general principal purposes is – Models built using just the content of the text in documents yields better prediction accuracies when compared to models built using both content and structure of the text in documents.

C. Future Directions

While we were able to derive certain general principles of feature selection for automated evaluation of descriptive answers, further research is required to apply sophisticated techniques such as Principal component analysis and perform feature transformation to verify if the model's performance can be improved. Yet another perspective in feature selection is to consider only the n-grams that appear in at least more than one document and verify if the model's performance is increased in that case. The philosophy here is that extreme rare features do not contribute well for the score prediction task performed by models, however this philosophy needs to be validated through formal experiments.

REFERENCES

[1] Brank J., Grobelnik M., Milic-Frayling N.,Mladenic D., "Interaction of Feature Selection Methods and Linear Classification Models", Proc. of the 19th International Conference on Machine Learning, Australia, 2002.

[2] Han X., Zu G., Ohyama W., Wakabayashi T., Kimura F., "Accuracy Improvement of Automatic Text Classification Based on Feature Transformation and Multi-classifier Combination", LNCS, Volume 3309, Jan 2004, pp. 463-468.

[3] Lindsay I Smith, "A tutorial on Principal Components Analysis", Feb 2002, pp. 13.

[4] Zu G., Ohyama W., Wakabayashi T., Kimura F., "Accuracy improvement of automatic text classification based on feature transformation", Proc: the 2003 ACM Symposium on Document Engineering, Nov 2003, pp.118-120.

[5] Syed M. Fahad Latifis et al., "Towards Automated Scoring using Open-Source Technologies", Annual Meeting of the Canadian Society for the Study of Education Victoria, British Columbia, 2013, pp.13-14.

[6] Platt, John (1998), "Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines", CiteSeerX: 10.1.1.43.4376