Proceedings of 2012 4th International Conference on Machine Learning and Computing
IPCSIT vol. 25 (2012) © (2012) IACSIT Press, Singapore

Improving HCCA using Automatic Summarization

Seemab Latif [1+], Mary McGee Wood [2] and Goran Nenadic [3]

[1] Dept. of Computer Software Engineering, MCS, National University of Sciences and Technology (NUST), Islamabad, Pakistan
[2] Assessment21, Sheffield, UK
[3] School of Computer Science, University of Manchester, UK

Abstract. Assessment is an important and widely used element of the taught education. Computer-Assisted Assessment (CAA) systems are helping teachers by automatic marking of assessments of low-stake high volume exams (e.g., school level). For accurate and efficient marking of the exams, these systems require either training on pre-marked data or creating detailed representation of templates of the model answers. Because of these reasons, CAA systems are not adequate for high-stake low volume assessments (e.g., university level). As compared to fully automatic assessment approaches, semi-automatic approaches are encouraged. Human-Computer Collaborative Assessment (HCCA) is one such approach that involves human marker in the marking process. HCCA clusters short text answers and then present these clusters to the human marker for marking. Clustering of long text answers is problematic because of problems pose by natural language. In this research, we suggest automatic summarization as a means to reduce noise and length of the text answers to improve clustering of answers and hence marking.

Keywords: Text Mining, Automatic Text Summarization, Document Clustering

1. Introduction

Assessment is a common and important component of taught education. It is primarily used to gauge how much a student has learnt and provide a measure through which external bodies can assess progress and compare different students. Unfortunately, setting and marking of assessment is a time consuming and often tedious process. In the era of automation, CAA systems aim to overcome the burden of manual setting up of exams and the marking process. CAA systems can be used to take some of the drudgery out of the assessment process. CAA systems usually test the knowledge skills of the students rather than understanding. This is because of the frequent use of Multiple Choice Questions (MCQs), which are supposed to assess the lower level of concepts. Use of MCQs can be de-motivating for students as researchers agree that MCQs focus on testing the surface level of students' knowledge. Commercially available CAA systems have little or no support for essays or long answers in assessment and/or have other disadvantages.

Essays and longer answers are written in a natural language. As such, there are many problems posed by natural language for fully automated assessment tools. These problems include spelling mistakes, understanding of the text and handling homonyms. An approach to deal with the problems of natural language is to use a Human-Computer Collaborative Assessment (HCCA) approach, which aims to exploit the strengths of the computer and the human marker in the assessment process [1]. For example, the human marker can handle trickier ambiguities of natural language while the machine is used for the repetitive elements. Rather than aim for fully automated marking, the idea behind HCCA is to let the computer mark what it can and pass the rest to the human marker. The idea behind the HCCA method is to cluster short text students' answers and then present these clustered answers to the marker for marking. For essay- type and constructed questions, document clustering can be problematic because students might end-up writing sentences and words that are not relevant to the question being asked or the topic of the essay. These

+ Corresponding author. Tel.: +923235555451.
E-mail address: seemab@mcs.edu.pk.

sentences and words are potentially noise for the document clustering and may lead to faulty clustering results. The document clustering process can be more efficient and accurate if noise is removed from the documents and the length of the document is reduced. One possibility to reduce the noise and the length of the documents is to extract the information content of the documents using automatic text summarization. We are proposing automatic text summarization as a method to pre-process documents and filter out the noise. In this method, students' answers are first summarized using automatic summarizer; then summarized students answers are clustered based on their similarity with each other using document clustering algorithms and then finally presented to the human marker. The human marker then marks each cluster of answers rather than each individual answer. Because of the involvement of the human marker, the complexities of natural language can be handled more accurately and efficiently.

The aim of HCCA is to use the computer to improve the efficiency and accuracy of the marking process. One way to achieve accuracy is to use document clustering to cluster similar answers into groups and then mark these groups of answers. Improving and speeding up the clustering process can achieve efficiency and further accuracy. Automatic text summarization can be used to summarize the students' answers and hence reduce the length of the answers and the irrelevant information.

2. Document Clustering with Text Summarization

Document clustering is the process of grouping similar documents such that similarity between documents within a cluster is maximized and similarity with documents of other clusters is minimized. Automatic text summarization is the process of generating a short coherent summary of a document using a computer such that the resultant summary contains the most important information of the document. Traditionally, documents are pre-processed before clustering using stopword removal, stemming, case folding and spell checking. All these pre- processing methods do not remove noise from the documents and reduce the length of the document. Document clustering becomes inefficient when the number of documents to be clustered is large (for example, k-means clustering) and when the length of the documents is long (for example, agglomerative clustering). In this paper, we aim to investigate the effects of document summarization as a pre-processing technique for document clustering.

3. Evaluating Clustering Time Efficiency

This section describes the experiment that aim at reducing the length of documents through automatic summarization as a pre-processing technique before document clustering. This experiment is about evaluating the performance time of the clustering. The use of basic pre-processing techniques does not give rise to the time efficiency of document clustering algorithms. One obvious reason for this is that document clustering algorithms are dependent on the length of the document [2].

3.1. Experimental Design

This experiment was conducted to observe the effects of summarization as a pre- processing method on document clustering. This effect was measured as the performance time of the clustering, which includes document pre-processing time, time for similarity calculation between documents and time to create clusters. Documents were represented as vectors of words in a VSM and time was calculated in milliseconds.

Setup: Two clustering algorithms were used in this experiment, namely k-means and agglomerative. The experiment was run on an unloaded, DELL Intel Centrino DUO T2300 @ 1.66 GHz machine with 512 MB RAM and Microsoft XP Home Edition operating system. The terms were counted in each document vector without and with summarization. If run separately, both automatic summarization and clustering have its own pre-processing methods. When integrated, we have further reduced the time by the performing pre-processing methods only once. The cluster generation time for both algorithms was measured. This cluster generation time included only the time to perform clustering on pre-processed documents that were represented in the VSM along with the similarity matrix.

Datasets: This experiment was performed on university examinations, which were taken from the ABC marking tool developed by Assessment21 (http://www.assessment21.com). These exams are from three different courses (course A has 3 exams, course B has 1 exam and course C has 2 exams). We have used fourteen question-answer sets; refer to as datasets, from these six examinations. Table 1 gives the question ID, along with number of answers and average number of terms in each dataset. These terms were automatically extracted, which represents the number of dimensions in the VSM. Clustering was performed on each of the fourteen datasets separately using both k-means and agglomerative clustering. The number of clusters in each case depends on the marks each question carries. If a question is worth 5 marks then the number of clusters for this question-answer set is 6 (one for un-attempted and 0 mark answers).

3.2. Experimental Results

The results for this experiment were calculated separately for the number of terms (36% decrease), pre-processing time without and with summarization (36% decrease), similarity calculation time without and with summarization (52% decrease), and total processing time for both clustering algorithms, when applied on full documents and summarized documents (increased by 85%).

Table 1: Datasets Statistics for Evaluating the Performance time of the Clustering Process

Dataset	Question ID	No. of Answers	No. of Terms	No. of Clusters
Course A	A1	280	708	6
	A2	279	960	6
	A3	271	1592	10
	A4	276	1012	6
	A5	276	944	7
	A6	441	1236	8
Course B	B1	113	668	3
	B2	110	705	5
	B3	108	880	5
Course C	C1	160	1593	8
	C2	165	940	4
	C3	145	857	5
	C4	139	1103	7
	C5	74	668	5

Table 2: (a) Average K-means Clustering Time (ms), (b) Average Agglomerative Clustering Time (ms)

(a)

Question ID	K-Means Clustering Time		% Decrease
	With Summarization	Without Summarization	
A1	256437.0	57532.0	77.56%
A2	251399.9	126739.2	49.59%
A3	683614.1	470178.1	31.22%
A4	345395.2	140221.5	59.40%
A5	311657. 8	138806.6	55.46%
A6	979468.0	646016.0	34.04%
B1	34018.7	14331.3	57.87%
B2	14262.4	7846.8	44.98%
B3	12345.3	6150.0	50.18%
C1	8479.4	4413.4	47.95%
C2	5781.2	2875.1	50.27%
C3	15695.4	8577.9	45.35%
C4	8167.3	3401.9	58.35%
C5	20369.3	11970.1	41.23%
		Average	**49.52%**

(b)

Question ID	Agglomerative Clustering Time		% Decrease
	With Summarization	Without Summarization	
A1	7862.7	6781.1	13.76%
A2	7459.3	6198.3	16.91%
A3	6726.4	5594.0	16.84%
A4	7243.8	6407.9	11.54%
A5	7139.2	5901.0	17.34%
A6	25031.2	24409.4	2.48%
B1	546.9	484.3	11.45%
B2	505.8	451.6	10.72%
B3	469.1	412.4	12.09%
C1	1318.8	1067.2	19.08%
C2	1448.3	1204.6	16.83%
C3	979.8	862.3	11.99%
C4	853.1	722.0	15.37%
C5	115.3	93.5	18.91%
		Average	**13.95%**

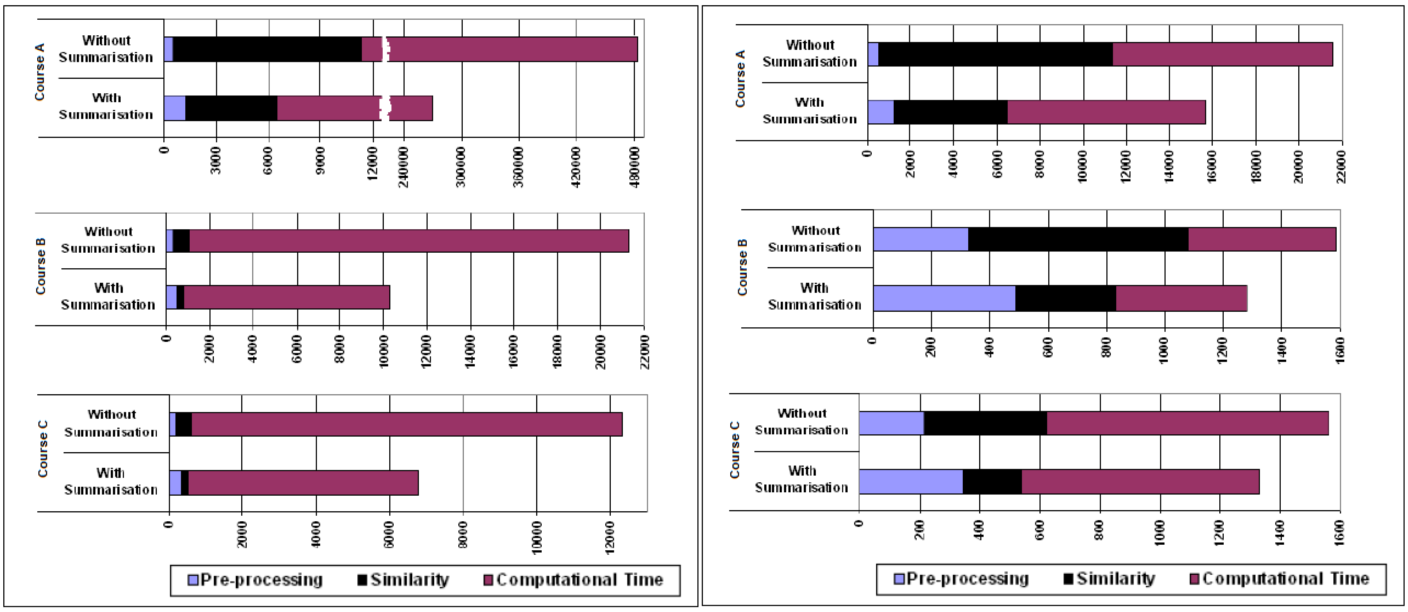
K-means and Agglomerative clustering algorithms were run 10 times on each dataset with and without summarization and then average values were calculated for further processing. Average clusters creation time of the k-means clustering algorithm when applied to full documents and summaries of the documents is shown in Table 2 (a). There is, on average, a reduction of 50% in clustering time. Table 2 (b) shows that the reduction in number of terms in the documents has very little effect on the agglomerative clustering. There was only 7% reduction in average cluster generation time for the summarized documents.

4. Analysis

Compared to agglomerative clustering, the traditional k-means algorithm is inefficient while working on large numbers of datasets and improving the algorithm efficiency remains a problem. Figure 1 (a) show that there is a significant decrease of 49% in the performance time of k-means clustering when it is applied to the summaries of documents as compared to the performance time of the algorithm when applied on whole documents. This means that k-means is sensitive to the number of terms in the documents. Analysis also shows that the total pre-processing time has increased but the overall time of the algorithm and the time to measure similarity has decreased. Overall, the average processing time for clustering each dataset using k-means is given in Table 3, when applied to full documents and summarized documents.

Table 3: Overall processing time averaged over each datasets for K-means

Dataset	Summarization	Pre-processing	Similarity	Clustering	Overall	%Decrease
Course A	Without	532.2	10814.5	471328.6	482675.3	
	With	1256.1	5210.0	263248.9	269715.0	44.12%
Course B	Without	326.0	754.1	20208.8	21288.9	
	With	487.7	341.1	9442.7	10271.5	51.75%
Course C	Without	213.5	406.2	11698.5	12318.2	
	With	344.7	195.0	6247.7	6787.4	44.90%



(a) **(b)**

Fig. 1: Overall pre-processing time for k-means (a) and Agglomerative (b) for three datasets

The overall computational time for the three datasets (averaged over the answers sets in each dataset) using k-means clustering and agglomerative clustering is given in Figure 1, Table 3 and Table 4 respectively. For k-means, Course B dataset has a 52% decrease because this dataset has only three question-answer sets with, on average, 110 answers in each set. Summarization as a pre-processing step has affected the Course B dataset clusters, using k-means, more than Course A and C clustering. In case of Agglomerative clustering, the Course A dataset is more affected by summarization, when it is clustered using the agglomerative algorithm.

Table 4: Overall processing time averaged over each datasets for Agglomerative

Dataset	Summarization	Pre-processing	Similarity	Clustering	Overall	%Decrease
Course A	Without	532.2	10814.5	10243.8	21590.5	
	With	1256.1	5210.0	9215.3	15681.4	27.3%
Course B	Without	326.0	754.1	507.0	1587.0	
	With	487.7	341.1	449.0	1278.0	19.47%
Course C	Without	213.5	406.2	943.1	1562.7	
	With	344.7	195.0	789.9	1329.7	14.91%

Summarization has shown very little effect on the cluster generation time of agglomerative clustering, but due to the reduction in pre-processing time and similarity calculation time there is an overall decrease of 32% in its performance time, shown in Figure 1 (b). The reason agglomerative clustering gains moderate effect from summarization is that the agglomerative algorithm is highly dependent on its implementation, in particular for pre-computation of the similarity matrix. Agglomerative clustering has an indirect effect from summarization, as the similarity calculation time is reduced to half after applying the summarization. Agglomerative clustering time is dependent on both the length of the document and the number of documents in the dataset, while k-means clustering is more affected by length of the documents in the datasets.

5. Conclusion

In this paper, experiment has been discussed which evaluates the method of improving clustering results by performing automatic summarization of students' textual answers. Automatic summarization has reduced the length of the documents and hence the number of feature terms that were potentially noisy for clustering. The results suggest that automatic summarization has filtered out the noise from the documents and has extracted the relevant information content of the documents. Due to the noise removal and reduction in document length, the performance of the two document clustering algorithms has improved in terms of performance time and quality of clusters. The time efficiency of k- means clustering has been reduced by 49% and that of agglomerative clustering is reduced by 32%.

6. Acknowledgements

I acknowledge the support of ABC Marking tool from Assessment21 for providing the exam data.

7. References

[1] J. Sargeant, M.M. Wood and S.M. Anderson. A human-computer collaborative approach to the marking of free text answers. In 8th International Conference on CAA, 2004 PP 361–370.

[2] C. Hsi-Cheng and H. Chiun-Chieh. Using topic keyword clusters for automatic document clustering. In the Proceedings of 3rd International Conference on Information Technology and Applications, IEEE Computer Society, 2005 PP 419– 424.

[3] Wood, M.M., C. Jones , J. Sargeant and P. Reed Light-weight clustering techniques for short text answers in HCC CAA. In 10th International Conference on Computer Aided Assessment, Loughborough, 2006 UK. PP 291–305.

[4] S. Latif and M.M. Wood. A Novel Technique for Automated Linguistic Quality Assessment of Students' Essays Using Automatic Summarizers. In Proceedings of IEEE Computer Society, 2009 PP 144–149.