Computer-Based Assessment: From Objective Tests to Automated Essay Grading. Now for Automated Essay Writing?

Robert Williams[1], John Nash[2]
[1]Curtin University of Technology, Perth, Western Australia, Australia
[2]University of Ottawa, Ottawa, Ontario, Canada
bob.williams@cbs.curtin.edu.au, nashjc@uottawa.ca,

Abstract. Assessment of student learning is an important task undertaken by educators. However it can be time consuming and costly for humans to grade student work. Technology has been available to assist teachers in grading objective tests for several decades; however these true-false and multiple choice tests do not capture the deeper aspects of student learning. Essay writing can be used to assess this deeper learning, which includes a student's ability to synthesize his/her thoughts, and argue for propositions. Automated essay grading systems are now starting to be used in the educational sector with some success. They can reduce the cost of grading, and they also eliminate the inconsistencies that are found amongst human graders when marking the same essay. The next development in essay processing technology is automated essay writing. This development will present a new set of challenges for educators. The detection of automatically generated essays may be difficult, and students may be given credit for writing which does not reflect their true ability. An understanding of how these systems would work, and the characteristics of the generated essays, is thus needed in order to detect them. This paper describes the components we believe an automated essay generator would need to have, and the results of building a prototype of the first of these components, the Gatherer.

Keywords: Automated essay writing, Automated essay grading, Gatherer.

1 Motivation for the Study

One of the authors (RW) has had extensive experience in building and testing an automated essay grading system. The other author (JN) thought about the logical extension of this technology to the automatic writing of essays. Subsequently one author visited the other for several weeks in 2008 during which his ideas were tested. This paper describes the background to essay writing and scoring, and the experiences with the prototype Gatherer system, the first component of the proposed automated essay writer.

2 Essay Scoring Technology

Computer based assessment began in 1955 when Lindquist developed optical test-scoring equipment at the University of Iowa. Large-scale testing programs, involving millions of students at all educational levels, are now commonplace. These programs are made efficient and effective through the use of computer and scanning technology [1]. This equipment however is only suitable for True-False and Multiple Choice questions, commonly known as Objective tests. Objective tests can measure many learning outcomes, however

> …there remain significant instructional outcomes for which no satisfactory objective measurements have been devised. These include such outcomes as the ability to recall, organize, and integrate ideas; the ability to express oneself in writing; and the ability to supply rather than merely identify interpretations and applications of data. [13].

3 The Value of Essays

According to Ebel essay tests

> …provide a better indication of students' real achievements in learning. Students are not given ready-made answers but must have command of an ample store of knowledge that enables them to relate facts and principles, to organize them into a coherent and logical progression, and then to do justice to these ideas in written expression. [9].

Essays also provide an indication of the nature and quality of students' thought processes, as well as their ability to argue in support of their conclusions [10]. The relative merits of Objective tests and Essay tests are summarized by Ebel as follows:

> An essay examination is relatively easy to prepare but rather tedious and difficult to score accurately. A good objective examination is relatively tedious and difficult to prepare but comparatively easy to score. [11].

We can conclude then that computer support for scoring objective tests is widely available, but that essay testing may be preferred for measuring the higher level abilities of students. If essays could also be graded by computers, then the time consuming tasks of human grading could be reduced and efficiencies in grading could be obtained similar to that obtained for objective tests. Computer grading of essays is now possible, and the accuracy of the grading can match that of humans. The question then arises as to whether students could obtain software tools to automatically write essays and fool the automated grading systems.

University students have always been required to write essays for assessment. An essay topic, expected length, and due date are generally specified by the lecturer. The student is then expected to research the topic, think about the issue, and write his/her

response. The student has to be careful about plagiarism, and to correctly reference source material. Essays are generally used when the lecturer wants to assess the student's ability to express and synthesize ideas, which cannot be measured by multiple choice or short answer tests.

4 Essays for Sale

Students today have available to them many World Wide Web (Web) sites that can provide an essay for a fee. Sites include

- Custom Writing: http://custom-writing.org/
- CustomEssays.co.uk: http://customessays.co.uk/
- Prime Essay: http://www.primeessays.com/
- Tailored Essays: http://www.tailoredessays.com/
- Order Papers.com: http://www.orderpapers.com/
- OvernightEssay.com: http://overnightessay.com/

These sites provide essays from databases of pre-written essays, or writers will write custom essays to order. Turnaround time can be as little as three hours. Detection of these bought essays is difficult because we assume that they are not published to the Web and hence cannot be detected by search engines.

5 Automatically Grading Essays

Essays can now be graded automatically by specialised software. We know of sixteen different systems, which are listed below.

1. AutoMark [20]
2. Bayesian Essay Test Scoring System [24]
3. Conceptual Rater [5]
4. Content Analyst [8]
5. Educational Testing Service 1 [3]
6. Electronic Essay Rater [4]
7. Intelligent Essay Assessor [15]
8. Intelligent Essay Marking System [19]
9. Intellimetric [27]
10. Blue Wren Software [2]
11. Paperless School Free Text Marking Engine [17]
12. Project Essay Grade [22][23]
13. Rx Net Writer [6]
14. SAGrader [14]
15. Schema Extract Analyse and Report [7]
16. Text Categorisation Technique [16]

These systems make use of natural language processing technology and statistical techniques to analyse style and/or content. Some of the systems typically use between fifty to four hundred human graded essays to train the systems for the specific essay questions. Multiple linear regression is often used to build a scoring equation from the linguistic and content features of these training essays. Ungraded essays are then assigned a score using this equation, and the specific predictor values for each essay. Most of these systems can perform as well as human markers in the sense that the computer-human score correlations are similar to the human-human score correlations on the same essays. The systems' computer-human score correlations tend to be between 0.70 and 0.90. One of the authors (RW) has developed an essay grading system [2]. One test of the system with several hundred essays of about four hundred words in length achieved a computer-human score correlation of 0.79 compared with the human-human score correlation of 0.81 [28]. These systems are starting to be deployed in primary and secondary schools, as well as universities. For technical details about some of the major commercial systems, and their performances, see [25] [26]. For critical evaluations of some of these systems see [12].

6 Computer Generated Essays

Essay processing technology is now starting to incorporate essay-writing systems. Perhaps the best known system is SCIgen (http://pdos.csail.mit.edu/scigen/), a system to randomly generate computer science research papers. A paper generated by the system was accepted as a non-reviewed paper at a conference in 2005. The question then becomes whether automated essay writing systems can generate intelligent and coherent essays which can fool university markers into assigning good grades to them. A second question is whether we can identify characteristics of automatically generated essays, and then flag the essays for the attention of the human graders. In order to understand this problem one of the authors decided to build an automated essay writing system and get a feel for these distinguishing features, if they exist. This system, GhostWriter, is currently under development.

We think an essay writing system should have the following functionality:

- A Gatherer to search the Web for documents relevant for the essay topic, retrieve these documents, and then assign a score for the degree of relevancy.
- The Organizer to select and assemble the appropriate sections of the retrieved documents which will form the body of the essay.
- Templates for defining the essential structure of the essay.
- The Compositor to build the essay from the retrieved material.
- A Spelling and Grammar Checker to standardize the grammar of the essay, and to correct spelling errors.
- A Reviewer tool that allows for quality checking.
- A Distortion module to mask the text copied from the Web documents in order to make the essay unique.

We have so far developed a prototype Gatherer, and in this paper we discuss its architecture, and the results of some testing we have performed with it. The Gatherer has utility outside automated essay writing, so is of value by itself as a form of meta-search tool.

7 Architecture of the Gatherer

The Gatherer takes as its input keywords that relate to the required essay topic. It also needs the user to specify search engine sites which will be used by the Gatherer to find the relevant documents. A simple Web page generated with some PHP scripting is used for obtaining the input controls i.e., the search terms. This information is then passed onto a Perl script that performs all the main required tasks and generates a simple Web page allowing the results to be accessed. This page is being enhanced to permit cleanup of temporary storage and better management of the searches. For example, it would be useful to be able to modify and rerun the search. Currently all keywords or phrases have the same weighting in terms of the search, and the system does not have a built-in Boolean logic ability to fine tune the query.

While the present proof-of-concept has deficiencies as mentioned, it has the particular strength of being compact and easily modified. We intend it to be an open-source tool and will shortly be making it available on the Web. Our approach is to use as far as possible the public face of search engines rather than their particular APIs (Application Program Interfaces). This reduces the risk that the API service will be discontinued (as several have been, e.g., Ask disabled their API in March 2007). However, the Web "face" of the search tools can change, and will force changes to our script.

We have attempted to structure the Perl script as a backbone with plug-in modules for each search tool and each document conversion. We intend to do the same for methods for scoring the retrieved documents, as ultimately we hope to be able to compare scoring strategies. We welcome collaborations.

8 Searching the Web

The following Web resources are currently used for searching:

- Wikipedia in English.
- Yahoo! News.
- Google (We are also considering the Google Scholar service, but are not sure this will remain open.).
- Ask.com.

It is envisioned that in the future other document repositories will be added to the system. For example, we have identified the Social Science Research Network

collection of papers (http://www.ssrn.com/) and the ACM Digital Library (http://portal.acm.org/dl.cfm) as potentially useful resources. The automated script for finding and then retrieving the Web documents uses the Web tool called WGET [21]. This method also saves the documents in files on the system's computer hard disk.

9 Document Formats

Common document types found on the Web include the following:

- HTML – hypertext mark-up language
- .txt – text file
- .pdf – Portable Document Format file
- .doc – Microsoft Word document
- .odt – Open Office Writer document
- .ppt – Microsoft PowerPoint presentation
- .odp – Open Office presentation

The system being discussed in this paper obviously needs to be able to process these multiple document formats, as a typical search will return a mixture of these file types. The system uses HTML as its base format for all processing. When .txt files are found, they are processed as though they were in HTML format. PDF files are converted using the open-source, cross-platform tool pdftohtml (http://pdftohtml.sourceforge.net/). Proprietary formats, such as Microsoft Word and Open Office Writer documents, can be converted to HTML using APIs present within these applications, and/or by third party tools. There may be issues with platform-independence with some of these document types however. For example, Microsoft Word documents cannot be handled directly by Unix\Linux platforms. Currently the system only converts PDF files, but the code is structured in a manner that lets us plug in other converters as needed.

Once the documents have been retrieved and converted to HTML, some editing of the documents takes place. Lowercase letters are converted to uppercase in search terms. Newlines tags are removed, multiple spaces are replaced with single spaces, and some special characters are also removed. Some simple analysis of the documents then takes place. We count the number of times the search word or phrase occurs, and take note of their positions within the document. At a future stage it is intended to make use of this information for scoring the relevance of the document, as our present algorithm is very crude and makes some obvious mistakes.

10 Observations, Ongoing Work and Conclusions

Our work in building a prototype Gatherer, and the testing of it we have undertaken, has indicated to us that the system is quite useful, not only as a component of the proposed GhostWriter, but also for other applications that require Web searches for documents on a particular topic. The Gatherer, in its current form, can be used not

only for finding the relevant documents – it also downloads them, saves them, and converts them into HTML. This is particularly useful for researchers, students, and people in industry who wish to prepare research reports on particular topics.

Proposed future development of the system includes a wider selection of Web sites for searching, the ability to convert documents to other formats than HTML, and to improve the platform independence of the Gatherer. We also want to build a better scoring algorithm which will indicate the relevance of the documents for the chosen topic. We also hope to build prototypes for the other components of GhostWriter.

Acknowledgements. The authors acknowledge the support of Curtin University of Technology for partial support for Professor Nash's academic visit in January and February 2008. Professor Nash is also supported by a grant in lieu of salary from the University of Ottawa during his sabbatical. The Telfer School of Management hosts the macnash server. Both Curtin University of Technology and the University of Ottawa have provided network and physical support allowing this work to proceed. Mary Nash kindly proofread a draft. Discussion and suggestions from Christian Guetl have been very helpful.

References

1. Baker, F. B.: Automation of Test Scoring, Reporting, and Analysis. In: Thorndike. R. (ed.) Educational Measurement, Second Edition, American Council on Education, Washington, D. C. (1976)
2. Blue Wren Software Pty. Ltd., http://trial.essaygrading.com
3. Burstein, J., Kaplan, R., Wolff, S., Lu, C.: Using Lexical Semantic Techniques to Classify Free-Responses. Proceedings from the SIGLEX96 Workshop, ACL, University of California, Santa Cruz (1996)
4. Burstein, J., Kukich, K., Wolff, S., Lu, C., Chodorow, M.: Enriching Automated Essay Scoring Using Discourse Marking. Proceedings of the Workshop on Discourse Relations and Discourse Markers, Annual Meeting of the Association of Computational Linguistics, August, Montreal, Canada (1998).
5. Burstein, J., Leacock, C., Swartz, R.: Automated Evaluation of Essay and Short Answers. In: M. Danson (ed.) Proceedings of the Sixth International Computer Assisted Assessment Conference, Loughborough University, Loughborough, UK. (2001)
6. California Electronic Writer/V.A.F., http://www.rxnetwriter.com/product_sheet.html
7. Christie, J. R.: Automated Essay Marking – For Both Style and Content. In: Danson, M. (ed.) Proceedings of the Third International Computer Assisted Assessment Conference, Loughborough University, Leicestershire, UK (1999)
8. Content Analyst Company, http://www.contentanalyst.com/solutions/essay.htm
9. Ebel, R. L.: Essentials of Educational Measurement. Third Edition, p. 96. Prentice-Hall, Englewood Cliffs, New Jersey (1979)
10. Ebel, R. L.: Essentials of Educational Measurement. Third Edition, Prentice-Hall, Englewood Cliffs, New Jersey (1979)
11. Ebel, R. L.: Essentials of Educational Measurement. Third Edition, p. 100. Prentice-Hall, Englewood Cliffs, New Jersey (1979)
12. Ericsson, P. F., Haswell, R. (eds.): Machine Scoring of Student Essays - Truth and Consequences. Utah State University Press, Logan, Utah (2006)

13. Gronlund, N. E., Linn, R. L.: Measurement and Evaluation in Teaching. Sixth Edition, p. 211. Macmillan, New York (1990)
14 Idea Works, http://www.ideaworks.com/sagrader/
15. Landauer, T. K., Foltz, P. W., Laham, D.: An Introduction to Latent Semantic Analysis, Discourse Processes, vol. 25, pp. 259--284 (1998)
16. Larkey, L. S.: Automatic Essay Grading Using Text Categorization Techniques. Proceedings of the Twenty First Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Melbourne, Australia, pp. 90--95 (1998)
17. Mason, O., Grove-Stephenson, I.: Automated Free Text Marking with Paperless School. In: Danson, M. (ed.) Proceedings of the Sixth International Computer Assisted Assessment Conference, Loughborough University, Leicestershire, UK (2002)
18. McGee, T.: Taking a Spin on the Intelligent Essay Assessor. In: Ericsson, P. F., Haswell, R. (eds.) Machine Scoring of Student Essays - Truth and Consequences. Utah State University Press, Logan, Utah (2006)
19. Ming, P. Y., Mikhailov, A. A. ,Kuan, T. L.: Intelligent Essay Marking System. In: Cheers, C. (ed.) Learners Together. February, NgeeANN Polytechnic, Singapore (2000)
20. Mitchell, T., Russell, T., Broomhead, P. Aldridge, N.: Towards Robust Computerised Marking of Free-Text Responses. In: Danson, M. (ed.): Proceedings of the Sixth International Computer Assisted Assessment Conference, Loughborough University, Leicestershire, UK (2002)
21. Nikšić, H., Cowan, M.: GNU Wget, http://www.gnu.org/software/wget/
22. Page, E. B.: The Imminence of Grading Essays by Computer. Phi Delta Kappan. January, pp. 238--243 (1966)
23. Page, E. B.: Computer Grading of Student Prose, Using Modern Concepts and Software. Journal of Experimental Education, vol. 62, pp. 127--142 (1994)
24. Rudner, L. M., Liang, T.: Automated Essay Scoring Using Bayes' Theorem. Journal of Technology, Learning, and Assessment, vol. 1 (2) (2002)
25. Shermis, M. D., Burstein, J. C. (eds.): Automated Essay Scoring - A Cross-Disciplinary Perspective. Lawrence Erlbaum Associates, Mahwah, New Jersey (2003)
26. Valenti, S., Neri, F., Cucchiarelli, A.: An Overview of Current Research on Automated Essay Grading. Journal of Information Technology Education, 2, pp. 319--330 (2003)
27. Vantage Learning. http://www.vantage.com/pdfs/intellimetric.pdf
28. Williams, R.: The Power of Normalised Word Vectors for Automatically Grading Essays. Journal of Issues in Informing Science and Information Technology, 3, pp. 721--729 (2006)