Concepts in Computer Aided Essay Assessment: Improving Consistency by Monitoring the Assessors

Richard V. De Mulder and Kees van Noortwijk
Erasmus University Rotterdam
Centre for Computers and Law
demulder@law.eur.nl / vannoortwijk@law.eur.nl

This paper focuses on a traditional educational skill, namely the assessment of student work. Whereas ICT has left a considerable mark on, for instance, the administrative support of educational activities and on the use of legal sources, other parts of legal training have seen almost no alterations in the past few decades. One such area is the grading of essay or open question student assignments. Although writing essays is an important aspect of student work, particularly in legal education, because of the necessity to train language skills, little has been done in this field. Now, however, a program is available to assist teachers in assessing and marking essays.

The CODAS Text Grader tool, described here, can be used to alleviate the task of marking essays. Does the use of this tool mean that a teacher can be replaced by a computer with respect to the marking of essays? The answer is negative; teachers still have an essential role. What changes is the 'level' at which the assessment of the student work takes place: from individual to survey, from marking to ranking. The software is capable of ranking essays based on concepts formed from a limited number of 'example documents' and helps to mark them. As a consequence, deviations become apparent immediately.

The CODAS software complies with the requirements of efficiency and reliability. It can also be used for a related task: it can assess the assessors. It contains functions to assess – and if necessary correct – the marks for comparable essays awarded by one specific teacher or even by a team of several teachers. Bringing transparency to the process of grading can only be of benefit to the education system.

Introduction

Like many other western countries, the Netherlands has been a knowledge economy for many years. Well-trained professionals are a must for this type of economy, therefore a growing number of students receive education for a longer period of time. Apart from this, the demand for education has increased as a result of globalisation. The number of students from China, Indonesia and Eastern Europe is already considerable in every Dutch university. The financial benefit universities receive from this has become a substantial part of their funding. The potential market share for traditional as well as distance education for foreigners is enormous.

As education is so important from an economic point of view and because competition between those who offer it has increased greatly, quality and efficiency have become of vital importance. An aspect of education that is crucial here is the assessment of student achievements. Assessment is important for the educational

process, as it provides feedback to students and teachers. Furthermore, reliable and valid assessment procedures are essential for the reputation of educational institutions and the certificates issued by them. In this paper, two new computer applications are described. These include an application that can be used by teachers to assess student assignments and an extension of this application that makes it possible to analyse and compare assessment results from different teachers and/or different groups of students (possibly from several different years) with each other and draw conclusions from the analysis.

Grading Assignments

Teachers usually consider the assessment and the grading of assignments, essays and examinations to be a burden. Consequently, they often rely on assignments that can be checked quickly or even automatically, for instance in the form of multiple-choice questions, especially when the number of students is high. Nevertheless, most teachers agree that essay questions are, in most cases, more suitable to gain insight into the knowledge and skills of students. This question type is usually considered superior for educational goals like 'improving skills to express oneself in writing'.

Until recently, automation in this area was only possible when using multiple choice questions or questions with very short answers. However, the Centre for Computers and Law at the Erasmus University has now developed a computer application that changes this. The Centre has been involved in research on the subject of reading and – to a certain extent – interpreting texts by computer for over fifteen years. This has been mainly aimed at the development of more effective, so-called conceptual retrieval systems for legal databases.[1] As a spin off of this research, an application that can serve as an aid in the assessment of (electronically supplied) essay assignments was developed some time ago. It has been in use for several years now and has been amended and improved in various ways since its original introduction.

The CODAS software

The CODAS software – CODAS stands for 'Conceptual Document Analysis System' – is based on the following principle: even though it is not yet possible to make a computer 'understand' the meaning of a text, it *is* possible to use a computer to analyse the *form* of the text, specifically the *word use* in it.[2] It has been shown that assignments that are similar at the word use level have, in most cases, a similar content as well. More specifically, student essays or open question assignments that show similar word usage also show similar content and receive comparable grades. This implies that there is a clear connection between the form of an assignment (expressed in word use characteristics) and its content or meaning.[3]

---

1 For a general introduction to conceptual retrieval of legal documentation, see Bing 1987. For an example of a practical application of such techniques, see Wildemast & De Mulder 1992 and De Mulder *et al.* 1993.

2 Van Noortwijk & De Mulder 1996.

3 Shermis & Burstein 2003; Combrink-Kuiters et al. 1999; De Mulder 1995.

This principle is utilized in the computer aided grading of assignments and examinations using CODAS. This does not mean that the grading process is completely automatic, however. The teacher is in control and determines the final results. How this works in practice is described in the following section.

Grading assignments in practice using CODAS

CODAS can be used for the grading of 'open' or 'essay' questions in an assignment or examination. Especially when the number of assignments is high (for instance, more than fifty), the amount of time that can be saved with this system can be significant. The grading process progresses in the following way.

Availability of assignments
For CODAS to be applicable, all student assignment must be available in electronic form, in other words, in the form of a computer file. Usually this can be achieved by requesting students to type in their assignments with a word processing application and to hand in the saved file. Handing in can take place by means of (attachments to) electronic mail, but it is usually more efficient to use a so-called 'Upload-script' (on a web-page) or a specific program. It is then possible to make sure that the name of every document that is handed in is unique, for instance by using the student's ID as the name of the uploaded document. Furthermore, repeated uploading of the same assignment can be prohibited and assignments can be grouped together in a specific directory on a network server, ready to be processed by the teacher.

The form of assignments
When this first hurdle has been cleared, the current version of the software usually requires one extra step to be taken before the actual grading can start. All assignments must not only be available in the original format of the word processing program, but also in a 'pure text' form. This usually makes it necessary to convert the documents. The program Microsoft Word contains a dedicated routine for this, which makes it possible to convert the whole series of assignments in one go. The Txt files that are the result of this step can be read by the CODAS software, whereas the original documents stay available to the teacher. These original versions are displayed whenever he/she requests the on-screen viewing of an assignment from the program.

Ranking the assignments
After the material has been prepared as described above, the CODAS modules can be applied.[4] Using the Text Grader module, the teacher can rank the assignments and award a grade to them. In this phase, it is essential that the teacher grades a series of assignments 'by hand' (although on the computer screen) and indicates examples of really good and really bad assignments. The program assists with this by calculating an *initial score* and provisionally ranks the documents according to this score.

[4] It is often useful to start with the separate CODAS Fraud Finder module that is capable of identifying possible plagiarism by students. In this paper we will not describe that module, as we concentrate on the grading process and its assessment.

The initial score

This first score is based purely on the similarity scores of the documents.[5] To calculate the initial score, the average is taken of the similarity scores of a document with all other separate documents. The idea behind this procedure is that on average, students will have found the correct answer. Therefore, an assignment that most resembles all the others will presumably be among the better assignments while an assignment that is dissimilar to all the others is probably one of the worst. In practice, this assumption has been found to hold better for the bad assignments than for the good ones. This implies that the worst assignments are usually indeed found at the bottom of the initial ranking, while the best assignments are not quite at the top of the list. Therefore, although not perfect, this initial ranking can be a useful aid to the teacher in his task of finding examples of good and bad assignments.

Locating examples and counter examples

Of the assignments that have caught the attention of the teacher because of their ranking in the initial score list[6], a selection is graded in the 'old fashioned' way. An assignment the teacher assesses, while searching for examples and counter examples is always awarded a grade. Furthermore, when indeed it proves to be a clear example of a good or bad assignment, it is marked with a '+' (for a good assignment or 'example') or '–' (for a bad assignment or 'counter example') respectively. After some (at least three) examples and counter examples have been identified, the 'recalculation' function can be activated. This produces a new 'grade score' for every document. This score, based on a statistical analysis of the word use, indicates the extent to which a document resembles the examples and is different from the counter examples. For the calculation of the score, Bayesian statistics are applied.[7] Therefore, the score of a document in the set represents the Bayesian 'odds' of that document, in this case equivalent to the probability that the document should receive a high grade. The list of documents is sorted according to this new score, which is visible in a separate column (see figure 1). This means that the ranking of the documents now reflects their calculated status. Generally, this ranking will become higher when a document contains a lot of words that can also be found in examples. It will become lower when the document has a lot of words in common with counter examples. In literature, this technique is often referred to as 'naïve Bayesian ranking' or 'naïve Bayesian retrieval', because of the implicit assumption that words appear independent of each other in a document. Although this is usually not the true for natural language texts, the results of the technique can be very convincing. This is illustrated by the fact that certain 'spam filter' applications for the blocking of unwanted e-mail messages are based on it.[8]

Applying grades

Of course, the teacher must now check the validity of this provisional ranking. This is achieved by opening and grading additional documents, at the top as well as at the bottom of the list. Usually, this will lead to the identification of additional examples and counter examples. Next, scores can be recalculated and changes in the ranking

---

5 See Van Noortwijk & De Mulder 1997.

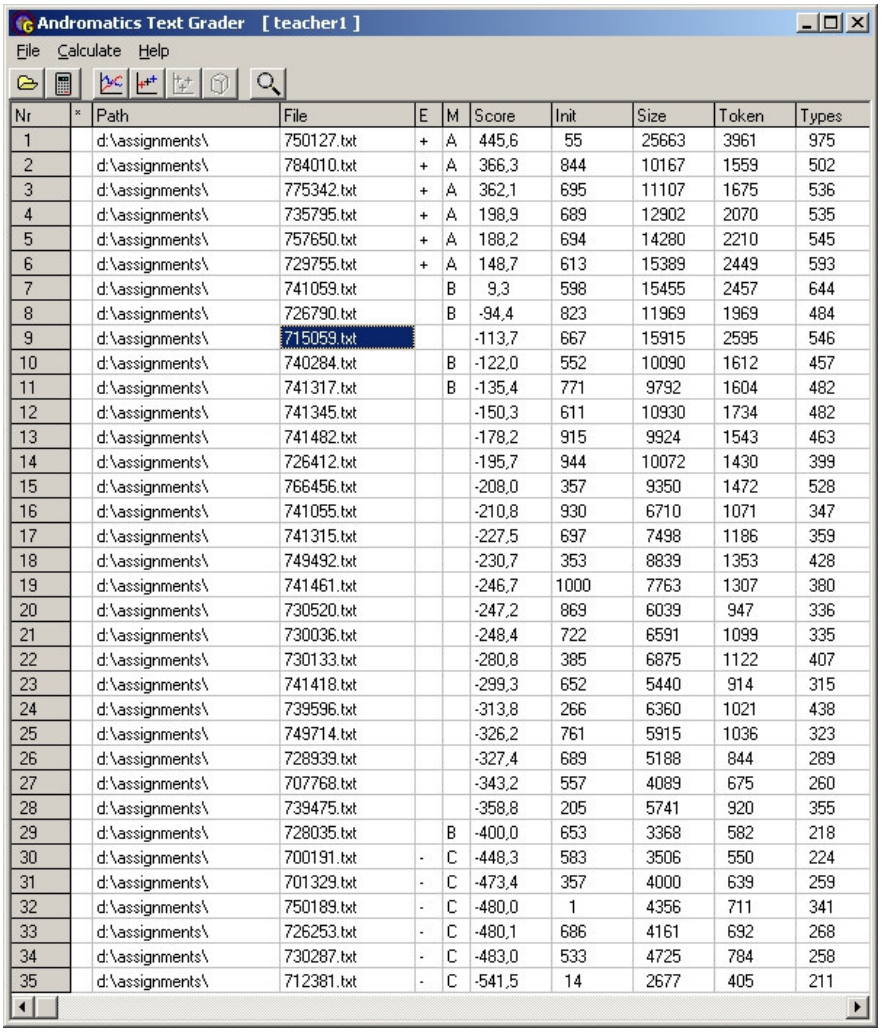
6 Another practical solution to the problem of quickly finding some 'good' and 'bad' assignments can be the addition of assignments from earlier years / courses with a grade that is already known.

7 A useful introduction to Bayesian statistics is given in Lindley 1971. There are many other examples of the application of Bayes' Theorem to the ranking and /or retrieval of documents. See for instance Elkan 1997, Mitchel 1997, Lewis 1998 and Eyheramendy *et al.* 2003.

8 For an evaluation of this, see Androutsopoulos *et al.* 2000.

of documents can be evaluated. This process can be repeated, until the teacher decides that the ranking is stable and forms a valid basis for the grading of the whole group. Finally, the borderlines between ‘good’, ‘sufficient’ and ‘insufficient’ (or any other desired level) assignments must be identified after which a grade can be applied to all remaining assignments.

The method works best when assignments have a certain minimal size, usually of at least some 500 words. This gives the software sufficient material to compare and to base the ranking on. Furthermore, it is desirable that the subjects in the assignments are not too divergent; preferably they should contain the answers to the same set of questions. When these requirements are met, the results are usually remarkably good.



Nr	*	Path	File	E	M	Score	Init	Size	Token	Types
1		d:\assignments\	750127.txt	+	A	445,6	55	25663	3961	975
2		d:\assignments\	784010.txt	+	A	366,3	844	10167	1559	502
3		d:\assignments\	775342.txt	+	A	362,1	695	11107	1675	536
4		d:\assignments\	735795.txt	+	A	198,9	689	12902	2070	535
5		d:\assignments\	757650.txt	+	A	188,2	694	14280	2210	545
6		d:\assignments\	729755.txt	+	A	148,7	613	15389	2449	593
7		d:\assignments\	741059.txt		B	9,3	598	15455	2457	644
8		d:\assignments\	726790.txt		B	-94,4	823	11969	1969	484
9		d:\assignments\	715059.txt			-113,7	667	15915	2595	546
10		d:\assignments\	740284.txt		B	-122,0	552	10090	1612	457
11		d:\assignments\	741317.txt		B	-135,4	771	9792	1604	482
12		d:\assignments\	741345.txt			-150,3	611	10930	1734	482
13		d:\assignments\	741482.txt			-178,2	915	9924	1543	463
14		d:\assignments\	726412.txt			-195,7	944	10072	1430	399
15		d:\assignments\	766456.txt			-208,0	357	9350	1472	528
16		d:\assignments\	741055.txt			-210,8	930	6710	1071	347
17		d:\assignments\	741315.txt			-227,5	697	7498	1186	359
18		d:\assignments\	749492.txt			-230,7	353	8839	1353	428
19		d:\assignments\	741461.txt			-246,7	1000	7763	1307	380
20		d:\assignments\	730520.txt			-247,2	869	6039	947	336
21		d:\assignments\	730036.txt			-248,4	722	6591	1099	335
22		d:\assignments\	730133.txt			-280,8	385	6875	1122	407
23		d:\assignments\	741418.txt			-299,3	652	5440	914	315
24		d:\assignments\	739596.txt			-313,8	266	6360	1021	438
25		d:\assignments\	749714.txt			-326,2	761	5915	1036	323
26		d:\assignments\	728939.txt			-327,4	689	5188	844	289
27		d:\assignments\	707768.txt			-343,2	557	4089	675	260
28		d:\assignments\	739475.txt			-358,8	205	5741	920	355
29		d:\assignments\	728035.txt		B	-400,0	653	3368	582	218
30		d:\assignments\	700191.txt	-	C	-448,3	583	3506	550	224
31		d:\assignments\	701329.txt	-	C	-473,4	357	4000	639	259
32		d:\assignments\	750189.txt	-	C	-480,0	1	4356	711	341
33		d:\assignments\	726253.txt	-	C	-480,1	686	4161	692	268
34		d:\assignments\	730287.txt	-	C	-483,0	533	4725	784	258
35		d:\assignments\	712381.txt	-	C	-541,5	14	2677	405	211

Figure 1 – Assignments with marks (A, B, C) and example status indication (+, –)

Other systems for the grading of essays

The subject of automatic essay grading has received considerable attention in the past few years. Educational Testing Service (ETS), an organisation for educational measurement, has played an important role here with the development of tools like

E-rater and C-rater.[9] Although the purpose of these systems might be similar to that of CODAS, the techniques that are used are often rather different. The C-rater application for the recognition of concepts in short-answer free responses, for instance, has to be trained to recognise all different forms and variations of every separate answer. The E-rater application is primarily intended for longer essays and evaluates these according to criteria like syntax, discourse, topical content and lexical complexity. Another application to assess essays automatically has been promoted by Landauer[10] and is based on so-called 'Latent Semantic Analysis', a technique to derive the meaning of words (in an essay) from the context they appear in. The use of Bayesian statistics, applied using example and counter example documents, is specific to CODAS. The algorithm used by the program has been fine-tuned for optimal results in an educational environment.

The traditional skills of the teacher

From the above sections, it will be clear that the CODAS software can by no means function autonomously. The teacher has an essential role. During the grading process, the examples selected by him or her are the single basis for the ranking and subsequent grading of the set of assignments. On several occasions, the system appeals to the specific skills of the teacher. This is the case at the beginning of the grading session, when a set of good and bad assignments must be indicated before the system can produce an appropriate ranking. In the phase that follows, the selection of examples and counter examples to be included in the set is of vital importance. The final result is completely dependant on this process. Finally, the teacher is the person who decides that the ranking of assignments is stable and valid and who specifies the grades that will be given to each of them.

This implies that 'computer aided grading' is by no means the same as 'grading by computer'. It is still the teacher who makes the decisions at every crucial stage in the process and remains responsible for the grading. There are some new skills involved in this process, however. Grading assignments and examinations no longer just implies 'Take the first one from the pile, read it, mark it and put it away'. Instead of this, the work is done at a different level. The teacher attempts to rank the whole set of assignments correctly. At least the very best and the worst assignments have to be identified. During the grading session, the set is reordered continuously. The session only ends when the ranking is considered to be stable and correct.

Assessing the assessors

The different 'level' at which the assessment of the student work takes place is even more apparent when more than one teacher is involved in the grading process of the group. Many teachers will recognize this type of situation from practice, for instance because they have been responsible for the establishment of a final grade from the grades of several other teachers. Due to the increasing importance, in society as well as economically, of education and therefore also of the assessment of student results, the number of organisations and institutions that have specific responsibilities

[9] See for instance Leacock & Chodorow 2003, Burstein *et al.* 2003.
[10] Landauer 2002, Landauer *et al.* 2003.

for the quality and efficiency of this assessment has grown considerably in the past decade. Computer aided grading can provide interesting possibilities for this specific task as well.

The CODAS Text Grader module already contains a number of facilities that enable the user to evaluate the consistency of the ranking of a series of assignments that has already been established by a teacher. For instance, when an assignment that has been marked with a + or – does not change position after this marking but instead stays in the middle of the ranking, this should raise suspicion. Also, large differences between the grading scores of example or counter example documents and the other documents that have a ranking close to these are an indication that the number of examples and / or counter examples is not yet sufficient. The Text Grader module contains some graphical tools to make phenomena like these clearly visible.

Another possibility offered by the module is to make the results of the grading work of the different teachers who have been involved in the assessment process visible, even if these teachers have not graded the same assignments. This will be illustrated by some examples.

The first example is rather trivial. When one of the assessors has marked an assignment as an example (+) and the other has not, the following pattern can become visible (see figure 2). In this chart, the assignments are shown as crosses. The colour of the first assessor (horizontal axis) is blue, that of the second assessor (vertical axis) is green. Completely yellow crosses without any blue or green lines represent assignments that have not been used as examples or counter examples.

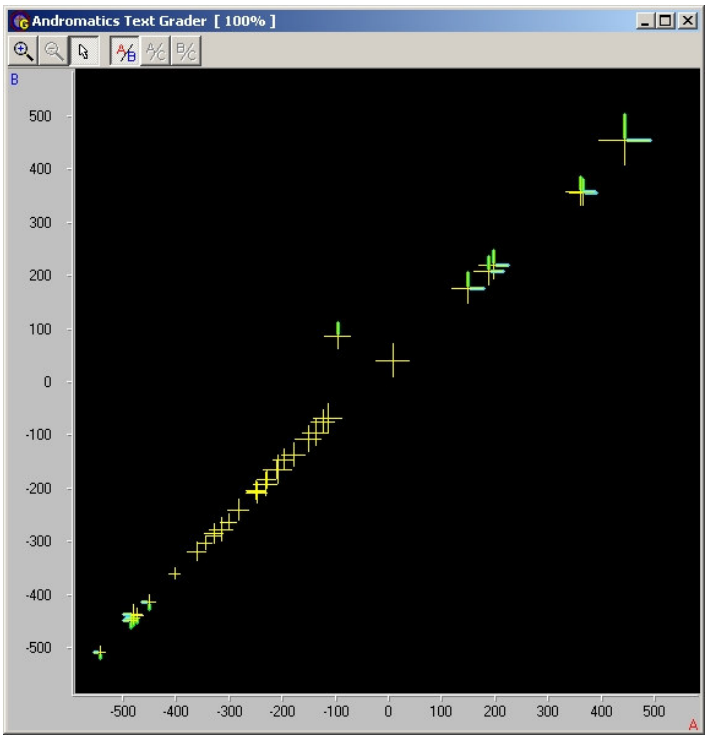


Figure 2 – Assessments of two teachers compared

The second example represents the case that both assessors apparently have a different opinion as to the quality of the assessment: what should be characterised as a good and a bad assignment (see figure 3). Their judgements vary considerably in a number of cases.

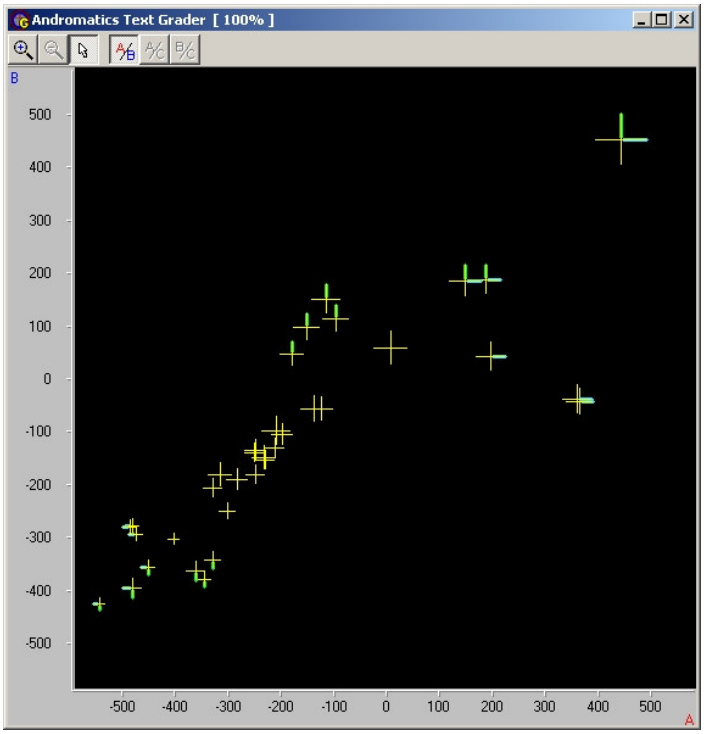


Figure 3 – Many differences between teacher A and B

Finally, it is also possible to view the coherence and consistency of the results of three different assessors in one chart (figure 4). In this case, the data points should appear more or less in one line or should be more or less 'cigar-shaped'. If not, one or more discrepancies between the assessors should be suspected. As this type of chart is sometimes a bit difficult to comprehend, the program contains options to rotate and/or tilt it to make it possible to judge the relative positions of the data points more easily. Furthermore, the 'shade' of the data points can be projected against any of the 'walls' of the chart (visible at the bottom of the 'cube' in figure 4), to illustrate the relationship between two of the three dimensions (teachers).

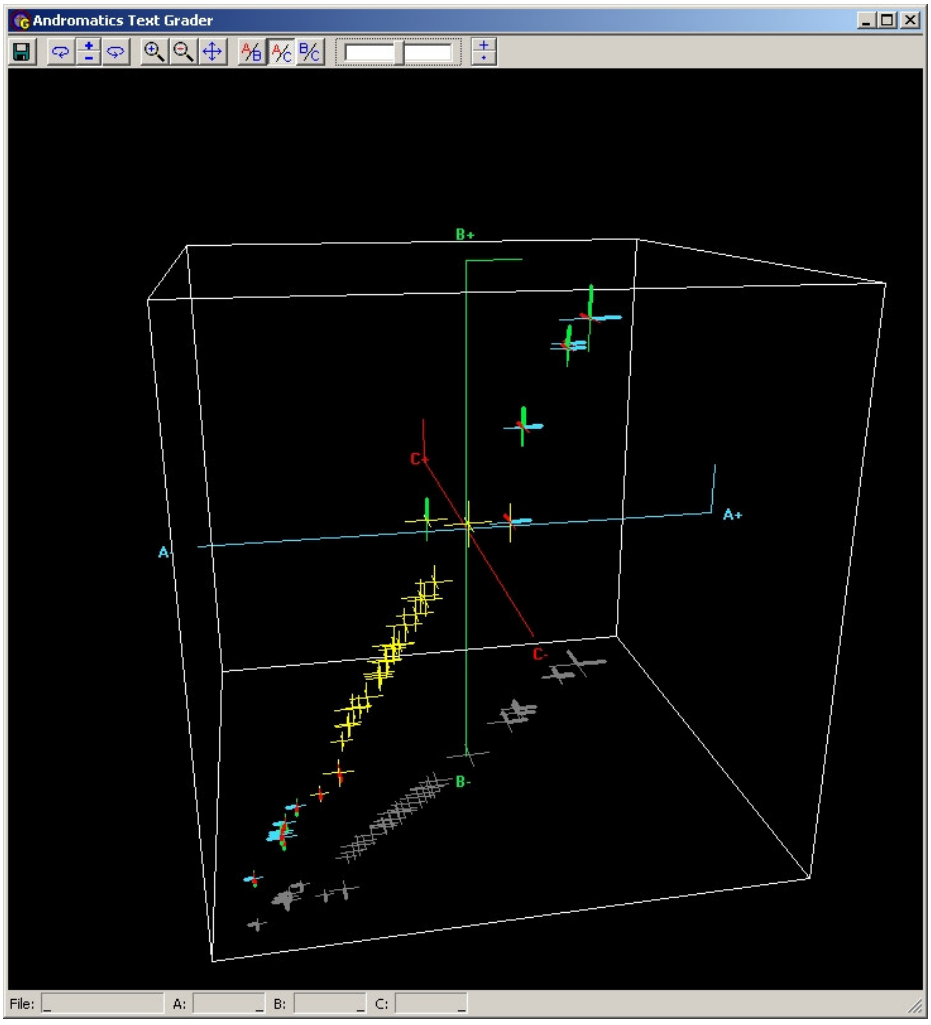


Figure 4 – Results of three teachers compared

Future developments

The assessment of assessors is not the main function of the CODAS software that is available at the moment. The two and three-dimensional charts were originally intended for the individual teacher to enable him or her to assess student work from different points of view and make those results visible. Nonetheless, the results of up to two colleagues can be reviewed, as described above. In the near future, this assessment functionality will be developed further. Options that are planned for this include the calculation of 'virtual' correlation scores between teachers, without the need for each of them to grade the complete set of assignments.

Another plan is to use the underlying CODAS algorithm for the identification of the 'author' of an assignment or essay. In a way that is similar to the way in which the ranking of assignments is based on examples by the Text Grader module, a list could be compiled that is ranked according to the probability that the documents are written by the same author. Such a module could be useful in distance education, where students produce materials without direct teacher supervision. The module would

complement the existing Fraud Finder module, which calculates the similarity of all pairs of documents in a set.

Furthermore, a number of users have asked for a 'feedback module'. Such a routine could retrieve the most relevant paragraphs (again, based on word use characteristics) from examples and counter examples, as indicated by the teacher. By comparing these paragraphs with any individual assignment from the set, students could be given relevant feedback on their work: 'An important issue that is missing in your assignment is …'. A prototype of this module has already been developed and is currently being tested in practice.

Conclusion

In this paper, some possibilities for using computers for the assessment of essays and assignments have been outlined. Assessment is a task that is considered to be increasingly important but is also very time consuming. The CODAS software assists the teacher in ranking a series of student assignments. Prerequisites for this are:

- the assignments must be available in electronic form (computer files);
- assignments must have a certain minimal size while the subjects they contain should not be too divergent;
- only a sample of the assignments needs to be graded in the 'old fashioned' way;
- the computer then ranks the assignments from good to bad;
- assessment becomes more consistent and reliable;
- considerable time savings are possible – the more assignments, the higher the time saving per assignment;
- the system makes it possible to compare results, between different years or different teachers.

The use of this tool requires, apart from traditional grading / assessing skills, also certain new proficiencies, for instance the ability to locate irregularities in the ranking. It is expected that computers, equipped with this type of software, will gain importance as a tool for checking and improving the quality of assessment in the near future.

Literature

Androutsopoulos, I, J. Koutsias, K.V. Chandrinos, G. Paliouras & C.D. Spyropoulos, 'An Evaluation of Naive Bayesian Anti-Spam Filtering', in: Potamias, G., V. Moustakis & M. van Someren (eds.), *Proceedings of the workshop on Machine Learning in the New Information Age*, 11th European Conference on Machine Learning, Barcelona, Spain 2000, pp. 9-17.

Bing, J.,'Designing text retrieval systems for conceptual searching', in: *Proceedings of the first international conference on Artificial intelligence and law*, Boston, Massachusetts, United States 1987, p. 43 – 51.

Burstein, J., M. Chodorow, & C. Leacock, 'CriterionSM: Online essay evaluation: An application for automated evaluation of student essays', in: *Proceedings of the Fifteenth Annual Conference on Innovative Applications of Artificial Intelligence,* Acapulco, Mexico 2003.

Combrink-Kuiters, Lia, Richard V. De Mulder, Henk Elffers & Kees van Noortwijk, 'Comparing Student Assignments by Computer'. In: *CYBERSPACE 1999: Crime,* Criminal Justice and the Internet, 14th annual Bileta Conference, York, 29th & 30th March 1999. Published in electronic form (CD-rom), 10 pp.

Elkan, Charles, *Naïve Bayesian Learning*, San Diego: University of California 1997.

Eyheramendy, S., Lewis, D. D., & Madigan, D. (2003). 'On the naive bayes model for text categorization', in: Bishop, Ch.M & Frey, B.J. (eds.), *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*, Key West, Fl: Society for AI & Statistics 2003.

Landauer, T.K., 'Applications of latent semantic analysis', in: *Proceedings of the Twenty-Third Annual Meeting of the Cognitive Science Society*, 2002, p. 44.

Landauer,T.K., D. Laham & P.W. Foltz, 'Automated Scoring and Annotation of Essays With the Intelligent Essay Assessor', in: Shermis & Burstein 2003.

Leacock, C. & M. Chodorow, 'C-rater: Automated Scoring of Short-Answer Questions', *Computers and the Humanities* 2003, 37:4.

Lewis, D.D., 'Naive (Bayes) at Forty: The Independence Assumption in Information Retrieval', in: *Proceedings of the 10th European Conference on Machine Learning*, London: Springer Verlag 1998, p. 4 – 15.

Lindley, D.V., *Making Decisions*, 2nd edition, London: John Wiley and Sons 1971.

Mitchel, T., *Machine Learning*, McGraw Hill 1997.

Mulder, R.V. De, M.J. van den Hoven & C. Wildemast, "The concept of concept in 'conceptual legal information retrieval'", in: *Proceedings of the 8th Bileta Conference*, 1st and 2nd April 1993, University of Warwick, Coventry 1993, pp. 79-91.

Mulder, R.V. De, 'Probabilistic approaches to legal concepts', in: Ciampi, C. *et al.* (eds.), *Verso un Sistema Esperto Giuridico Integrale*, Tomo I, Milano 1995, p. 125-140.

Noortwijk, C. van & R.V. De Mulder, 'Word use in legal texts: Statistical facts and practical applicability', in: Kralingen, R.W. van *et al.* (eds.), *Legal Knowledge Based Systems: Foundations of Legal Knowledge Systems* (Jurix '96), Tilburg: Tilburg University Press 1996, ISBN 90-361-9657-4, p. 91-100.

Noortwijk, C. van & R.V. De Mulder, 'The Similarities of Text Documents'. In: *JILT - Journal of Information, Law and Technology*, Issue 2/1997, 10 pp. University of Warwick, Coventry 1997.

Shermis, M.D. & J.C. Burstein (eds.), *Automatic Essay Scoring, a Cross-disciplinary Perspective*, St. Paul (USA): Assessment Systems Corporation 2003.

Wildemast, C.A.M. & R.V. De Mulder, 'Some considerations for the design of conceptual legal information retrieval systems', in: *Legal knowledge based systems, Information Technology and Law,* Jurix '92, Lelystad: Vermande 1992, p. 81-92.

CODAS on the internet: *http://www.andromatics.com*