Using freely available tools to produce a partially automated plagiarism detection process

Thomas Lancaster
School of Computing
University of Central England

Fintan Culwin
Faculty of Business, Computing and Information Management
London South Bank University

An increasing reliance on commercial tools for non-originality investigation of student submissions is taking academic integrity beyond its comfortable zone. There is no guarantee if such tools will continue to be available and, if they are, that they will be available at a reasonable price. Further the technical underpinning of such tools is unclear and has not been made publicly available. This may present problems if subsequently an academic misconduct investigation is started. Moreover, existing tools may not be the best suited to any given circumstance.

This paper describes a set of tools, all freely available, for finding similarity within corpora of student submissions and investigating what might, after due process, be termed plagiarism. The tools are described both technically, covering how they work, and operationally, identifying how they might be used as part of a defined plagiarism detection process. The development of the tools has taken into account the human aspects of running such a process; all employ visual techniques to aid in the investigation of non-originality. The tools are intended to help automate many of the areas that are already carried out by hand in academic institutions.

Keywords: Plagiarism detection, student cheating, interactive visualisation, human computer interaction

Introduction

One of the key roles for academic staff is to ensure that academic integrity is maintained within their institutions. One of the crucial requirements in order to maintain such integrity is ensuring that students who attain awards have achieved sufficient learning outcomes to merit them. It is the belief of many tutors that the problem of plagiarism is becoming endemic (Baty 2004). The problem of such high levels of plagiarism is two fold. One group of students are receiving awards that they have not fully earned. A second group of students who are producing work using legitimate means are being penalised. This is because their submissions may appear less impressive against the work submitted by the plagiarisers.

As the comfort zone is becoming breached, institutions are looking at both reducing the levels of plagiarism inherent within their course, as well as being able to demonstrate to the media and the wider academic community that they appreciate that they have a problem (Culwin and Lancaster 2001b). Two complementary schools of thought exist as to the best way to reduce plagiarism. The first relies of careful design of assignment specifications to minimise the chances for students to plagiarise, combined with innovative teaching methods that test different skills to those assessed by the traditional essay. The second relies on academics explaining to students what plagiarism is and why it is not acceptable, but relying on the students to preserve their own academic integrity. This option makes further use of tools designed to detect where students have colluded with each other, known as intra-corpal plagiarism, or copied from external sources such as the World Wide Web, known as extra-corpal plagiarism.

The tool that appears to be receiving the most media attention is iParadigms' tool TurnItIn.com (2004). This is a Web based service that allows students or staff to submit work in an electronic format, which is scanned against a database of Web pages and other documents for which iParadigms have negotiated access rights. Each assignment submitted can then be presented to a tutor in the form of an originality

report. This is an HTML formatted page containing a textual representation of the document, with sections of text replaced by hyperlinks to sources in the iParadigms' database. It is up to a tutor to then check each of these sources and to determine if these represent plagiarism or may instead represent acceptable similarity. A tutor can then use these tests and the results to put together some kind of report to be used in a plagiarism investigation. The originality report can be used to partially assist with this role.

Although TurnItIn.com (2004) receives the most media coverage of the available detection services this does not imply that it is the best or the most appropriate of the services, only that it is the one that has been the most aggressively marketed. Indeed as a commercial and marketed service it is essential for TurnItIn.com's economic success that the company maintains an appropriate market share, something that does not always sit comfortably with other anti-plagiarism alternatives. Those institutions that are integrating their courses with TurnItIn.com, especially now that the company is increasingly marketing itself as a complete educational workflow solution, are leaving themselves liable to future uncertainty.

This paper suggests that institutions look alternatively towards freely available open source software instead of the commercial plagiarism detection services. A number of free detection tools are available of which many have been developed by the Centre for Interactive Systems Engineering (2004), based primarily at London South Bank University. The tools cover the distinct stages of detection, where student submissions similar to one another or to external sources can be identified and the stage of verification, where a tutor can look at the results in order to make a judgement on whether to initiate formal investigation. These are two of the stages in Lancaster and Culwin's (2001) four stage plagiarism detection model. Although the tools are individually available they are being developed into a more coherent whole. Part of this work has already been completed. The intention is to integrate the remaining tools in the near future. They are further undergoing continual development and improvements as an influx of time and project students allows. The tools are available on the research centre's Web pages (CISE 2004).

This paper describes the main tools, both from a technical perspective and from the perspective as how each of them could be used as part of a recognised and defined plagiarism detection process operating mainly at a local departmental or institutional level. The tools cover identifying similar plain text submissions, investigating work showing signs of collusions and searching the Web to find potential plagiarism sources. The tools were developed using Java and are hence available over the Web as applets to be deployed under many operating systems and work environments.

Originality checker (OrCheck)

OrCheck is a tool that can be used as an interactive and visually assisted alternative to TurnItIn.com. OrCheck is used to automate the process used by a tutor where they believe a document to have been plagiarised from one or more Web sources. The tool covers two of the four stages of the detection process, namely detection and verification.

For the detection stage OrCheck presents a user with a filtered list of all the single words in a selected document and all of the three consecutive word phrases, with the 200 most common words removed from the lists. The user can then select a number of these terms to be automatically searched for using functionality from the Google APIs. Figure 1 shows an example of the three word term selection process on an essay on plagiarism largely copy and pasted from a number of sites offering detection advice.

OrCheck automates the process of searching for each of these identified terms, a very time consuming process when done by hand. It collects the top ten hits found for each word or sequence of words, or the number that exist if there are less than ten. Each of these is downloaded, either from the original HTML source where possible, or using a version stored inside the Google cache. A user can manually add additional search terms to check Google for, or additional sites if they already know some which are likely to have been used.

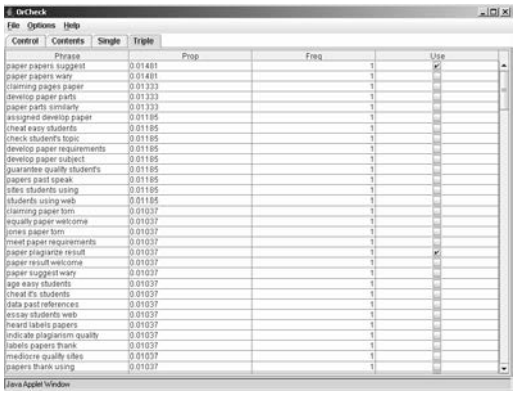

Figure 1: OrCheck term selection process

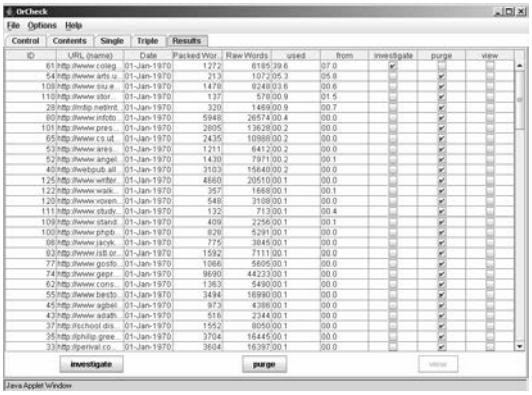

Figure 2: OrCheck used to identify sites from where work may have been copied

Each downloaded document is then automatically compared in more depth with the source document to give an approximate proportion of similarity between the two documents. The results of this comparison are shown in a table on a tab in the OrCheck window. Each line of the table lists a Web source, the proportion of the source document that has been found in the possible copy, as well as the proportion of the possible copy that might have been found in the source. These values might differ as the lengths of the respective documents are likely to be different, plus the student may have deployed disguise techniques. OrCheck uses a fuzzy matching algorithm so that it can see through some of these disguises. Figure 2 shows an example of the table generated.

For each document in the table three options are given about how the Web document could be used in the verification stage. These can be changed as required as the user expands their search. Those documents with a high level of similarity have the investigate tick box selected, this means they will be used in the verification stage. The remainder, with a lower level of similarity, have purge selected. Choosing the purge option will remove them from the system's corpus. Each document also has a view option to allow it to be seen in isolation. Usually the defaults represent a good indication of which documents are worth pursuing, as by its very nature checking out the top ten hits will lead to a number containing similar material at only a very superficial level.

Choosing the investigate option opens up a further tab which shows how the original source document compares with each of the Web documents. Unlike the TurnItIn.com view, which merely shows hyperlinks between the source and Web documents OrCheck also contains an interactive visualisation of similarity. The visualisation is kept deliberately simple and fuzzy so that the results of it can appear almost immediately to a waiting user. Figure 3 shows an example of the OrCheck visualisation and the associated screen produced when investigating a single likely source.

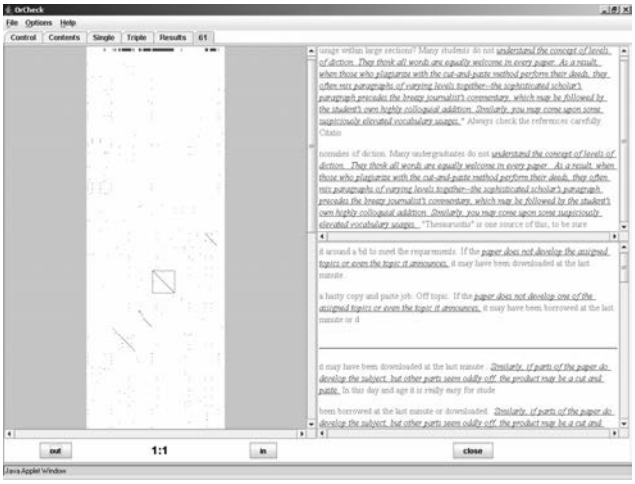

Figure 3: OrCheck visualisation identifies document copied from a web source

The OrCheck visualisation plots the student document across the x-axis from left to right. Each source is plotted, overlapping each other, down the y-axis from top to bottom. Where a word is identified common to the student source document and a Web document a coloured dot is plotted. A run of similarity between the two documents hence causes a diagonal line to be plotted of a single colour. The lines are interpolated across the x-axis at the top so the user can at a glance see the proportion of the document that student has submitted that is similar to each of the documents found during the assisted Web search. The length of the run is a good indicator of the extent from which the Web document may have been copied.

The diagonal lines are interactive, when the user moves the mouse over a near diagonal line a selection box is placed around it. Clicking the mouse shows the corresponding areas of the student source document and the Web document in a pane to the right of the main OrCheck window. A second pane at the bottom right shows all the common pairs between the source and other documents. These are also interactive providing an alternative to the visual view.

In the case of the document shown in Figure 3 the vast majority of the source document appears to have been copied from a much longer Web document. The sections of the Web document have been re-arranged, so that the earlier they appeared in the original the later they appeared in the copy. The third run of similarity is being looked at in the upper right pane and shows a close match with some level of disguise immediately before and after the run of similarity. The red line across the top of the visualisation shows that around 75% of the document has been copied from this single Web source.

OrCheck can be used as part of a detection process, most notably where plagiarism is already suspected in a particular document and a tutor wishes to largely automate the process of using Google to find and investigate possible sources. Culwin described one approach to using OrCheck to help educate first year Computing students (2004). OrCheck can also be used to check an entire set of student documents, hence ensuring that detection is non-capricious.

Plotted ring of analysed information for similarity exploration (PRAISE)

PRAISE is a tool designed to allow a tutor to investigate students who might have colluded with one another. As with OrCheck one of the strengths of PRAISE is the use of a new visualisation technique to show which documents are similar to which other documents and how similar those documents are. Two documents identified by PRAISE to be similar can be chosen by a tutor to investigate in further detail using the VAST tool, which is described in more detail later in this paper.

PRAISE is usually applied to students from a single course and for a single subject who might have committed intra-corpal plagiarism, that is students who may have copied from one another. The PRAISE corpus can be augmented with external sources, for instance Web sites or chapters of a set text that have been identified during an earlier procedure. Alternatively student submissions from the same module in previous years could be added to the corpus to allow students that have copied from previous work to be identified. PRAISE allows documents to be multiply linked to one another, which means if several students have worked together on an assignment specification they can all be clustered together during the automated detection stage of the process.

When it is first started PRAISE allows a user to select a set of documents to be examined and a metric to process them under. Additionally a base file, perhaps containing supplied materials, can be given. Sequences from the base file that exist within student documents will be ignored during processing. The latest version of PRAISE will also optionally extract all the urls from the corpus, download them and include them in the analysis. The reason for this is that one student may have cited a source that another student has copied from.

By default documents are compared at the word pairs level, the rationale for which forms part of Lancaster's PhD thesis (2003) and is further discussed by Lancaster and Culwin (2004). Other simple metrics, including runs of words or characters of different lengths, or comparisons at the sentence level, can also be selected; these might be more appropriate for specific subject areas.

As an example, the word pairs metric is calculated as follows. All the documents within the corpus are split into a list of all the word pairs contained within them. That is all the sequences of two consecutive

words, for instance “the cat sat on the mat” would produce “the cat”, “cat sat”, “sat on”, “on the” and “the mat”. Any capitalisation and punctuation that appears within these word pairs is eliminated and for the purposes of computation these are sorted alphabetically.

A similarity score, nominally scaled between 0 and 100, is produced for each possible pairing of documents within the corpus. The formula applied represents the proportion of pairs in common and is written as:

$$\frac{100(c1+c2)}{(c1+c2)+(u1+u2)}$$

here c1 is the common sequences from document 1 and c2 is the number of common sequences from document 2. Both these values will be identical, but are included within the formula for reasons of symmetry. Term u1 represents the number of unique sequences to document 1 and u2 shows the number of unique sequences to document 2. The denominator is, in effect, the total number of words in both documents. The values give an indication of the level (but not the percentage) of similarity between any pair of documents within the corpus but can also be used in visual form to identify clusters of similarity.

By default PRAISE uses this information to plot an interactive tork visualisation of the documents in the corpus. Figure 4 shows an example of the tork view produced on a corpus of 25 actual student documents.

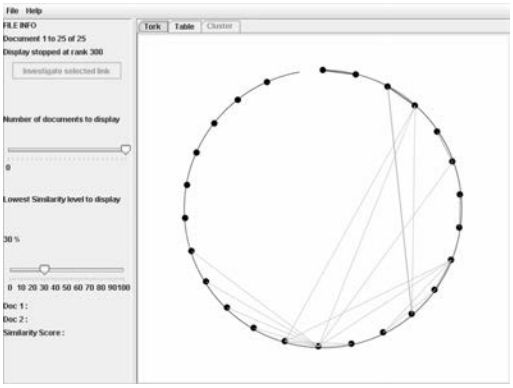

Figure 4: PRAISE tork view

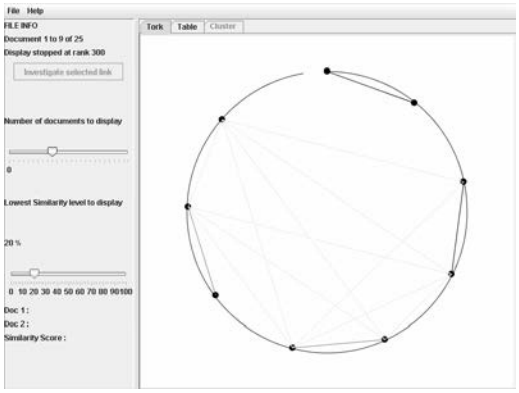

Figure 5: Less cluttered version of PRAISE tork view

The tork view sees each document plotted around the circumference of a circle, nominally started at the top and evenly spaced around the outside. Where a level of similarity has been detected between two documents a straight line is drawn. The thickness of the line is an indicator of the level of similarity. Further the position of the document around the circle from the origin position is an indicator of how much similarity the document shares with the corpus as a whole. Hence the uppermost document is often the one that a user will wish to consider first.

A user can select two linked documents by clicking on both their end points. This displays information about the extent of the similarity in the pane to the bottom left of the window. It also allows the selection of an investigate document button, which will open the two documents in the VAST tool so that the verification stage of the four stage process can be carried out. Alternatively a user can deselect one of the documents and select another to find out more about it.

If the view is too cluttered the tutor using PRAISE has access to two sliders at the left of the window. The first one can be used to reduce or increase the number of documents displayed around the circle, removing those initially that are judged least similar. For large corpora this can help the view become more manageable. The second slider allows the user to alter the level of similarity beyond which connecting lines are plotted. One use of this is to allow a user to focus in on the most similar documents. Figure 5 shows the same corpus as Figure 4 but with the view changed to include only nine documents and the minimum similarity level reduced to empathise the effects of clustering.

The user can switch out of the tork view to one of two more traditional views shown in tabs across the top of the PRAISE window. A table view plots the similar pairs from most to least similarity with the nominal similarity score shown, an example of this is shown in Figure 6. Note that these values give solely a representative score, rather than a percentage, due to the process of computing the similarity metric. A tutor might find the ordering from the most to least similar pair more useful. A cluster view can also be selected for an individual document, showing all the highly similar pairs that that document is involved with. Figure 7 shows an example of the cluster view for the document positioned to the NE of Figure 5. The investigate link is also available from these alternative views.

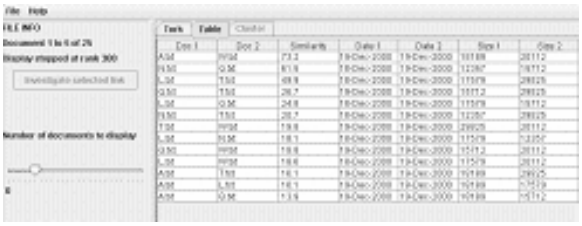

Figure 6: OrCheck table view (shown truncated)

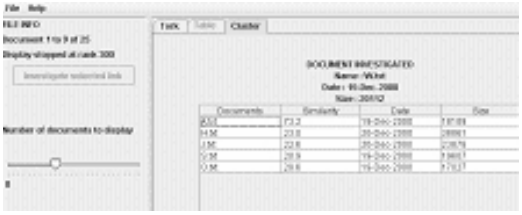

Figure 7: OrCheck cluster view (shown truncated)

Generally PRAISE has been found to be useful for checking for collusion, or for comparing documents to a database of possible sources, such as those identified by an OrCheck assisted search. Operational use of PRAISE might involve preparing a corpus of documents, investigating the most similar pairs and stopping when time constraints make further verification impractical or when the tutor is satisfied that they have found the majority of undue similarity within the corpus.

Visualisation and analysis of similarity tool (VAST)

The traditional method deployed by a plagiarism detection tool in the verification phase is simply to present a pair of hyperlinked documents to a tutor. It is then the job of the tutor to look through these linked areas and work out whether or not they are similar and if they are similar to work out what the relevant proportion and location of similarity is. This is not always an easy job when the documents span many pages.

The VAST tool (Culwin and Lancaster 2001a, Lancaster and Culwin 2001b) is intended to allow for a much refined and improved version of this process. As with the remainder of the suite of tools it incorporates a novel visualisation. This visualisation could be considered to be a much more detailed version of the OrCheck visualisation which uses fuzzier techniques for matching areas of similarity.

VAST is primarily used to investigate two documents that have been flagged as similar in some way, but where the reason for the similarity is not immediately intuitive. It can also be used to investigate a student source document against external sources. A VAST process can be spawned directly from PRAISE or the tool can be used independently.

The VAST view consists of a window with panes on the right containing the two documents under consideration and the panes on the left containing an interactive visualisation of similarity between the two documents alongside some background information about the documents. Due to the more detailed nature of the visualisation the process of generating the VAST visualisation is more dependent of computational resources than the other visualisations. When ready the visualisation shows a 2 Dimensional representative plot of the documents. The plot is interactive, the internal selection tool can be moved to identify areas of the visualisation that are interest and the system will navigate the two documents to the approximate linked areas. This allows the documents to be quickly compared for the purposes of investigating localised similarities. Figure 8 shows an example of the VAST view for the two documents judged most similar under the PRAISE view.

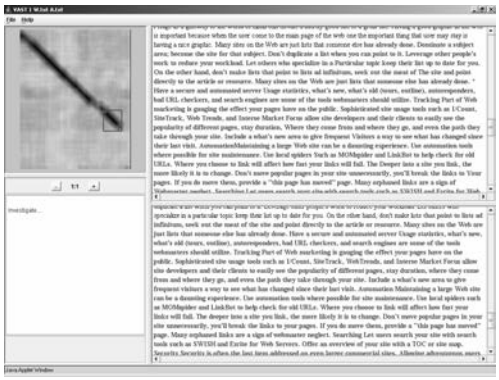

Figure 8: VAST used to investigate two highly similar documents

Figure 9: VAST used to investigate two less similar documents

The strength of any tool of this type is in the visualisation used within it. Hence at this point technical details of how the VAST visualisation is produced are useful.

Consider two documents: the first of word length X, the second of length Y (with X and Y both at least 200). VAST will generate a visualisation that is (X-200)/10 by (Y-200)/10 pixels, each rounded down to the nearest integer. So comparing two 1000 word essays would generate an image 80 pixels square.

Given that the origin (0,0) is in the upper left, pixel (x,y) is generated by comparing the run of words (10x+1) to (10x+200) of the first document with words (10y+1) to (10y+200) of the second document, conditional to 10x+1<=X and 10y+1<=Y. Hence pixel (0,0) is generated by comparing the first two hundred words of each document. Pixel (1,0) is generated by comparing words 11 to 210 of the first document to the first two hundred words of the second document.

Nominally 200 is known as the fragment size, this effects the level of fuzziness allowed in the image and was chosen as a result of comparative experimentation. The relatively large value takes into account properties of information retrieval that state that larger word samples are better at identifying areas of gross similarity. The value of 10 is known as the fragment gap. This largely affects the size of the VAST visualisation produced, downsizing it from comparing every possible fragment combination of 200 words, to one in every 100 (10 squared) fragment combinations. This was chosen to give a good balance between clarity and processing times. Note that there is an overlap between the way that successive pixels are generated which has the effect of giving a graduating effect to the visualisation.

The detail of how the intensity of each pixel is produced is also of interest. A choice of two metrics is available within VAST, each generating a value between 0 and 200, which are scaled to a greyscale value proportionally between white (0) and black (200). The unsequenced visualisation is generated by simply counting the number of words in common between the fragments of 200 words from each document. The sequenced visualisation is generated by calculating the length of the longest common substring common to both documents. That is it finds the longest possible run of words that are in the same order within the two fragments, but not necessarily including every word, or requiring that the words are located consecutively. The resulting visualisation will contain intense areas, known as similarity intersections, where there is a large amount of similarity between the two documents.

The sequenced and unsequenced metrics are largely trade offs against one another. The sequenced gives a clearer indication of similarity, since background noise has less of an impact. Such noise, a result of those function common words in the English language such as ‘the’ and ‘and’, clearly happens less frequently when considering ordering than by simply counting the number of shared occurrences. There are two trade offs against this clearer visualisation. The unsequenced visualisation method is able to cope much more satisfactorily when sections of the documents have been re-ordered within a single 200 word fragment. The unsequenced visualisation is computationally faster to generate and hence involves the user waiting for less time. Despite these trade offs informal testing generally finds the clearer unsequenced visualisation to be the more suitable of the two, although this could be domain dependent.

The view in Figure 8 shows two highly similar documents, where the documents are likely identical apart from some minimal attempt at disguise. This is reflected by the intensity of the intersection and the fact it runs nearly down the entirety of the primary diagonal. The positioning of the intersection is unsurprising as most copying appears to happen around such points, however a strength of the VAST visualisation that it also allows intersections that are positioned around the extremities to be quickly verified.

Figure 9 shows the second most similar pair of documents from within the corpus. The similarity shows rearrangement of two largely similar areas under the unsequenced metric. The selection tool can be quickly located over the sections of interest to verify if they contain indications of plagiarism or not.

During the verification phase the VAST visualisation can be zoomed, to give more detail about an area of the documents, or to give more of an overview, however this is only a digital zoom so overall detail will not be improved. When zoomed to frame the whole visualisation the overview allows a tutor to easily see what proportion of the documents are similar to one another and hence to quickly decide whether or not these documents are worth investigating further. The clear similarity intersections make the verification of the possibly plagiarised areas quick and easy, in order for a tutor to investigate if they represent collusion or merely correctly cite the same common source.

Generally the VAST visualisation seems to see through areas of attempted disguise more readily than OrCheck due to the more fuzzy nature of its matching algorithm. It also provides a more detailed view. VAST is particularly useful during the plagiarism detection process when looking for intra-corpal plagiarism, due to the integration of the tool with PRAISE. It has also proven useful for tutors wanting to easily find out how documents are similar and for tutors wishing to put a case together to clearly demonstrate similarity in a visual way for an investigation into academic misconduct.

FreeStyler

The final visual aided tool currently available from CISE is FreeStyler. The purpose of the tool is markedly different to the others. Instead of finding, verifying and investigating similarity in a corpus of documents FreeStyler is intended for use on only a single document at a time. FreeStyler is a tool designed to analyse the stylistic properties of a document in order to find out if it was written by a single author. As such areas identified as being stylistically disparate to the remainder of a document might be plagiarised and can be used for a Web search, or as evidence to require a student to undergo a viva.

The FreeStyler interface presents a document in a lower pane and a number of linked visualisations in an upper pane. Each visualisation takes the form of a graph of rolling averages of some property of a single document, in much the same way as VAST plots comparative properties of multiple documents. A further pane contains general information about the document being examined.

The FreeStyler visualisation panes are interactive, that is a tutor can use the selection tool to choose a start and an end point. The tool will automatically highlight an approximately corresponding textual area for examination. Moving the selection tool in one visualisation pane will move it across all graphs, this allows areas that are disparate from the rest of the document in multiple graphs to be easily identified.

A series of different metrics are implemented in FreeStyler, each of which will generate an individual visualisation. As with VAST FreeStyler is produced using a defined fragment size and gap which are the results of internal experimentation. The values used are a fragment size of 50 words and a gap of 5 words.

The details of the calculation process are as follows: Given a document of word length X, where X is at least 50, FreeStyler will generate a visualisation that is (X-50)/5 pixels wide, with height rescaled automatically so that the highest point equals the height of the window and other points scaled proportionally below.

Assuming that the origin is to the left of the pane (an x-coordinate of 0), the y-coordinate at position x is generated by taking the sequence of words with a sentence starting nearest after or including word (5x+1) to the sentence ending nearest after or including word (5x+50), processing them under the current metric, then scaling. Hence the y-coordinate at x-coordinate zero is generated from words 1 to the end of the sentence at word 50. The y-coordinate at position 1 is generated from sentences approximately starting

and ending from word 6 to 55. In effect, for speed of calculation, one in five possible points for the graph is sampled.

Some of the metrics available include the reading age (under Flesch-Kincaid or Fog), mean word length, mean words per sentence, mean characters per sentence, the type of voice, plus user definable letter sequences (e.g. the distribution of 'the' throughout the document).

Figure 10 shows FreeStyler being used to investigate one of the documents from the corpus already considered under OrCheck and VAST. The metrics displayed are the Fog score and the Voice score, largely representing the changing form of the reading age and the passiveness of the voice respectively. The section under investigation was chosen due to a dip in the Fog score at that point and a corresponding start of a change in the voice level. The section looks as if it may have come from a Web site.

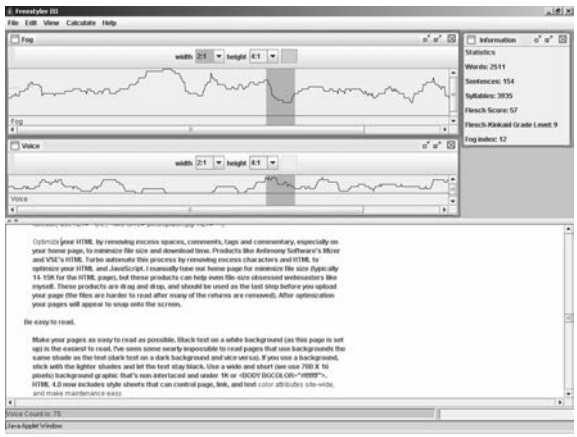

Figure 10: FreeStyler used to investigate a student document

Operationally FreeStyler has been used with more success on longer documents with relatively simple authorship style changes, for instance where the majority of the document is written in a style that is either largely linearly consistent or highly inconsistent, whereas the remainder is highly variable or highly consistent respectively. It is believed that the areas where several of the visualisations show a variance from the norm are the ones that are most worth testing, although this has not been formally investigated. As part of a detection process FreeStyler is a useful tool when a tutor believes that something is wrong but cannot quite put their finger on it, or cannot identify a source.

One alternative use of FreeStyler is to allow individual authors to check the consistency of their writing style. For instance, a reading age graph could be plotted to see if the document's readability matches expectations throughout, or if there are areas of inconsistencies that could be improved. It is hoped that this might also prove to be a useful tool for students looking to improve their academic writing style.

Eliminating the comfort zone

Although the evidence is largely anecdotal it appears that students are becoming increasingly comfortable with cheating; for many of them it could be a procedure they have learned at school and continued with into higher education. As it becomes more and more important for academic institutions to have a recognised plagiarism detection and procedure in place the tools available from CISE can be used to help ensure that academic integrity is maintained (CISE 2004).

Once documents are available in an electronic form as part of the collection phase of the four stage detection process the OrCheck and PRAISE tools are freely available to aid with detection. The appropriateness of each tool depends on whether intra-corpal or extra-corpal plagiarism is being searched for. Additionally external sources collected from OrCheck can be added to the corpus used by PRAISE. The verification phase can be aided using either the OrCheck visual view, or checked more thoroughly using VAST. The same tools can also be used to aid with putting together a report for the final stage of investigation. Documents can also be examined using FreeStyler to find stylistic inconsistency and this can be used to feed in to OrCheck where appropriate.

Although the tools represent an aid to the detection process, especially with the novel visualisation processes deployed, they are not yet a complete solution. Rather it is important that institutions and departments have a plagiarism detection policy in place that encourages tutors to automatically check student submissions for non-originality as a matter of course, rather than just assuming that students will not cheat. It is also important for tutors to be clear that their students understand what plagiarism is. Otherwise it is possible that some students will have cheated unintentionally.

The most recent problem hitting institutions experienced by the authors is that of students who are paying other people to do work for them. This is something that will never be flagged by plagiarism detection tools since it represents original work that is simply not the students own. The growing available of sites such as RentACoder (2004), Experts Exchange (2004) and Google Answers (2004) provide a ready made community of people who will complete assignments or put together programs at a price far less than the equivalent effort that a student would have to exert to do the work for themself. As a result institutions need to be looking at ways of ensuring that work submitted is the student's own. Possible solutions include examinations or vivas. However it should also be possible to build up a stylistic database of student writing styles, based on work produced in class or of known authorship. A tool like FreeStyler could be augmented with this information to provide a further solution to the newest problem in plagiarism detection.

References

Baty, P. (2004). Survey shows cheating is rife. *Times Higher Education Supplement*, 2 July 2004.

CISE (2004). http://cise.lsbu.ac.uk/tools.html [viewed 12 July 2004].

Culwin, F. (2004). An active introduction to academic misconduct and the measured demographics of misconduct. At Plagiarism: Prevention, Practice and Policy Conference, Newcastle, UK.

Culwin, F. & Lancaster, T. (2001a). Visualising intra-corpal plagiarism. *Proceedings Information Visualisation 2001*, London, UK. [abstract only, verified 30 Oct 2004] http://csdl.computer.org/comp/proceedings/iv/2001/1195/00/11950289abs.htm

Culwin, F. & Lancaster, T. (2001b). Plagiarism issues for higher education. *Vine*, 123 (1), 36-41.

Experts Exchange. http://www.experts-exchange.com/ [viewed 12 July 2004].

Google Answers. http://www.google.com/answers/ [viewed 12 July 2004].

Lancaster, T. (2003). *Efficient and Effective Plagiarism Detection*. PhD thesis, London South Bank University.

Lancaster, T. & Culwin, F. (2001). Towards an error free plagiarism detection process. At ITiCSE 2001, Canterbury, UK.

Lancaster, T. & Culwin, F. (2004). A visual argument for plagiarism detection using word pairs. At Plagiarism: Prevention, Practice and Policy Conference, Newcastle, UK.

RentACoder (2004). http://www.rentacoder.com/ [viewed 12 July 2004].

TurnItIn.com (2004). http://www.turnitin.com/ [viewed 12 July 2004].

Thomas Lancaster, School of Computing and Information, University of Central England, Perry Barr, Birmingham B42 2SU, UK. thomas.lancaster@uce.ac.uk
Fintan Culwin, Faculty of Business, Computing and Information Management, London South Bank University, Borough Road, London SE1 0AA, UK. fintan@lsbu.ac.uk

Please cite as: Lancaster, T. & Culwin, F. (2004). Using freely available tools to produce a partially automated plagiarism detection process. In R. Atkinson, C. McBeath, D. Jonas-Dwyer & R. Phillips (Eds), *Beyond the comfort zone: Proceedings of the 21st ASCILITE Conference* (pp. 520-529). Perth, 5-8 December. http://www.ascilite.org.au/conferences/perth04/procs/lancaster.html