

A REVIEW OF ELECTRONIC SERVICES FOR PLAGIARISM DETECTION IN STUDENT SUBMISSIONS

Fintan Culwin
Centre for Interactive Systems Engineering
School of Computing
South Bank University
London SE1 0AA
fintan@sbu.ac.uk
<http://www.scism.sbu.ac.uk/~fintan>

Thomas Lancaster
Centre for Interactive Systems Engineering
School of Computing
South Bank University
London SE1 0AA
lancaste@sbu.ac.uk

ABSTRACT

Student plagiarism is an ever-increasing problem for academic institutions. A growing number of students are using material from the Web in their submissions, without properly acknowledging the source. This paper reviews the need for widespread plagiarism detection systems and evaluates available Web based detection services. Four services are discussed: the Measure of Software Similarity (MOSS) service for program source code and the plagiarism.org, Integriguard and copycatch.com services for free-text submissions. The downloadable Essay Verification Engine (EVE) tool for free-text detection of Web plagiarism is also evaluated. The paper finds that all five could be invaluable resources for academic institutions as they strive for a pro-active anti-plagiarism policy. The paper concludes by looking at the authors' current work to combat plagiarism.

Keywords

Plagiarism, cheating & copying, forensic linguistics

1. OVERVIEW

Plagiarism (the presentation of another person's ideas or material as if it were one's own) has always been an issue of concern to academic institutions. Recently, the number of plagiarism incidences reported by the media has increased and it is likely that other incidents are going undetected by institutions that do not like to think that they have a problem. Extensive plagiarism undermines the value of qualifications awarded by an institution and infuriates those students who gain qualifications by legitimate means. Hence effective anti-plagiarism policies and systems are essential to guarantee

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

8th Annual Conference on the Teaching of Computing, Edinburgh

© 2000 LTSN Centre for Information and Computer Science

academic quality.

Plagiarism itself is not a new concern. In 1995, Culwin and Naylor outlined the problem and described a software system to check for similarities in program source code submissions [2]. More recently World Wide Web sites have appeared that claim to look for similarities in free-text submissions. It can be said that these are a necessary evil; the growth of the Web has provided students with access to large amounts of source material that they can *copy and paste* directly into an essay. Whole sites exist where complete assignments can be downloaded and handed in by students as their own work [4].

Such *Web plagiarism* is an increasing complication to tutors, already watching for duplicated material in student submissions. Many students realise that the likelihood of being detected is low, since work is often spread between many markers who are unlikely to mark a pair of directly plagiarised submissions. As academic institutions are run more like mass industries, tutors who would previously have known their students cannot be expected to learn the details of so many students' writing styles. The ease with which an essay or section of a Web page can be loaded into a word processor and reformatted also makes it harder for tutors to notice duplicate material unaided.

This paper looks at the increasing amount of academic resources available to aid tutors detect plagiarism. Support for finding similarities in both the (long established) constrained text and the (more recent) free-text areas of writing are examined. The paper ends with a look at the future of anti plagiarism measures.

2. CONSTRAINED TEXT

Until relatively recently, the processing power of computers has made it difficult to search for plagiarism within anything other than constrained text. As a result there are plenty of systems to find similarities in the major type of constrained text, program source code. Verco and Wise reviewed many of these systems [5] and Wise developed his own YAP3 system for just this purpose [6]. Culwin

and Naylor also developed a TEAMHANDIN system [2].

This paper is not a technical review of how such systems work, but they will usually tokenise a source code submissions in such a way, that, for example, renaming a variable throughout, or splitting a print statement over two lines, can be detected.

A more recent addition to the field is the Measure of Software Similarity (MOSS) service, operated by the University of Berkeley [10]. This allows tutors to collect a corpus of source code submissions locally and then submit it to the MOSS server to be checked. The source code would usually be collected through a local Web interface, to minimise clerical support time, but could be collected on disk, or by other methods.

The MOSS service is currently free to instructors on programming courses (although this may only be for a limited time) and setting up a new account is straightforward. The system works with C, C++, Java and ML source code, all of which are commonly taught languages in undergraduate programming courses. The instructor can supply source code expected to be common to each submission (e.g. supplied starter source code) and this can be discounted. The sensitivity of the analysis can also be controlled.

After processing, the server sends an e-mail giving the location of a Web page where the results can be found. Each pair of submissions within the corpus is ranked and the tutor can compare submissions side by side, with those parts of the submissions that MOSS believes to be similar colour coded. The tutor can then decide whether to take the case further.

The MOSS service has been in use at South Bank University, and other UK institutions, for some time with successful results.

3. FREE-TEXT

Looking for similarities between free-text submissions (most notably essays) has only become feasible in the last couple of years, with dramatic drops in the price of processing power and memory. There are now three Web sites and a downloadable tool designed to look for plagiarism in student submissions.

Denhart [3] compared plagiarism.org [11] with the downloadable Essay Verification Engine (EVE) [8] by submitting an essay made up of sentences from four well-known authors, with some amount of paraphrasing and re-ordering. He found that plagiarism.org decided that the essay was original, but EVE found most of the plagiarised material on the Web.

This section repeats the process with a carefully constructed sample text, designed to test the systems. The sample starts with two sentences each from Charles Babbage [13] and James Gosling

[14], both famous works widely available on the Web. In addition it contains a paraphrased sentence from an essay on the Year 2000 (Y2K) problem, obtained from one of the free essay sites [12] and a short sentence from a newly indexed personal Web site.

The services compared are plagiarism.org and EVE, as above, plus Integriguard [9] and copycatch.com [7], both looking for the same market as plagiarism.org. This section looks primarily at how effective each is at identifying the plagiarised material, but also considers how usable each is for tutors and students alike.

3.1 Integriguard

Integriguard [9] provides a ninety-day free trial of their service, after which instructors are charged 59.40USD per year. The trial account is straightforward to set up, giving each supervisor a unique number, although the password is displayed on-screen in unencrypted form, which is not ideal. After that the instructor simply gives their identification details to the students, who must register on an associated student site. Students then submit their work by copying and pasting it into a text box. A few minutes later the results of the analysis of that submission are e-mailed to the tutor.

The service claims to test each submission by comparing it against its database of writings. The returned e-mail showed that all six sentences of the sample text had been considered, and that none of them were in the Integriguard database. However five of the six sentences were accompanied by addresses of Web locations, where duplicates of the material can be found. Located sentences are given individually and with their surrounding context so that tutors can decide if they are properly referenced.

The sites found were of interest. For Babbage the search engine found the original site, a second site that quotes a small section of Babbage in agreement and, inexplicably, an adult entertainment site. Sentence two, also from Babbage, found the same three sites and an additional one. Material from the much-reproduced Gosling paper was found on eleven sites and nine sites respectively. Sentence five, a sentence from the Y2K paper that was actually in context a legitimate quote from a Y2K expert, found the original source for the quote, as well as ten essay sites which stored the paper. The final sentence, from the recently published personal Web site, was found to be original.

The whole process seems generally unsatisfactory, the presentation on the site and in each e-mail is poorly laid out and there is no guarantee that the work students are submitting to Integriguard is the same work they independently submit to the departmental office. Web sites are listed across multiple lines, meaning that a user has to cut and

paste the pieces into a browser to check out the source material. No effort is made to distinguish legitimately cited material and so a tutor would need to be very wary before starting proceedings against a student based on the evidence returned by Integriguard. The sample checked is also likely to miss small amounts of plagiarised material. This method would also generate a lot of e-mails for a tutor with a large class.

3.2 Plagiarism.org

Plagiarism.org [11] allow users to submit up to five papers as part of a free trial, initiated by supplying an e-mail address and filling in a questionnaire. Unlike Integriguard, plagiarism.org requires an instructor to give their address, telephone number and academic affiliation during the registration process. It also freely displays the password being chosen. The (text format only) file can be submitted by way of a text box or file upload. Students register and use a separate site as part of the paid-for service, which could not be tested. Tutors produce a separate submission area for each assignment they set.

Site design is generally effective and seems more up-to-date than Integriguard (dated 1998). It is also more commercially motivated, providing information as to what plagiarism is and links to press sites praising the plagiarism.org service. The system claims to work by comparing each submitted file to its database, using its own registered algorithm, as well as searching the Web. If a match is found details of a tutor with a matching assignment are said to be returned, the paper itself is not released.

Results from the service came after about 24 hours, via an e-mail with a pointer to a Web page. The paper was flagged as having some plagiarised material, presenting the entire submission on screen, with sections of it hyperlinked to source material on the Web. The system found the entire section of Babbage, linking back to our original source and the entire section of Gosling, linking to a different site. It also found the paraphrased section of the Y2K report, highlighting only those parts that remained unaltered and linking to one of the essay sites. Plagiarism.org flagged the material from the personal Web site as original.

The presentation of both the site and the plagiarised results is very good, although there is a notable delay in waiting for the results. The plagiarism.org service seems to work well, but would be unusable for most academic institutions, the price being 1USD per uploaded paper.

3.3 Copycatch.com

Copycatch.com [7] is the newest of the Web based plagiarism detection services. It is currently supported by advertising on the student site and hence provides a free service. It checks that only

tutors are using the service, contacting the departmental office before allowing an instructor to register, a process that can take up to two working days. Much like the other services, the system checks papers against its internal database and the wider Web. Unlike the other services, submissions can be made in many common word processing formats as well as plain text, although this doesn't take away the need to hand in a local, printed copy. However the submission system refused to accept the test essay in plain text format. A version encoded in Microsoft Word format was accepted successfully.

Once registered as an instructor, a tutor can add details of a new assignment, including a cut off date, to the system. Tutors then pass on identification details to their students, who have to submit their papers to the system before the cut off. There is no leeway to this date, late papers will simply not be checked for plagiarised material. When the deadline is reached copycatch.com processes the corpus and e-mails the tutor with a list of the number of lines in each essay and the number of lines matched. More detailed results are made available on the copycatch.com website for 48 hours.

The results are presented much like those of plagiarism.org, with the textual representation of the paper presented in a frame, with sections highlighted as html links to indicate a web source. The tests found parts of all four sections, as well as unrelated parts of the linking text, where the similarity was purely coincidental. Most of the Babbage paper was found at the original source, although for some reason one word was not highlighted. The section from the Gosling white paper was found, but as part of a set of lecture notes at a US institution. Sections of the Y2K paper were found on two different sites (even though the entire text was present on both sites). One of the located sites was where the original author had posted the information (which then found its way onto various essay sites). The system also found part of the material from the personal Web site.

Copycatch.com seems to be the most useable of the three services, although there is little to choose between any of them in terms of results. At present, being free, copycatch.com wins hands down on value for money.

3.4 Essay Verification Engine

The Essay Verification Engine (EVE) [8] unlike the other services listed is not Web based. Instead it is a downloadable tool (cost 19.99USD with a limited free trial) which checks the contents of an essay against some of the Internet search engines. It is designed mainly to be used when Web plagiarism is suspected, to aid a tutor in discovering the source.

By means of a simple wizard interface, users select files that they believe may contain plagiarised

materials. These can be in Microsoft Word, RTF or plain text formats. They can also select the strength of the test as high, medium or low. A medium strength test took about ten minutes on a fast machine, requiring continual access to the Internet.

The results are presented in a pop-up window on screen, with sections of the essay highlighted, along with matching Web links. These can be saved as an RTF file, or printed out for future reference, although in these latter cases it is not always easy to see which sections of text are linked with which Web site.

The results from EVE are the least convincing of all the services, finding plagiarised material in only the Babbage document and the Gosling Java white paper. For each of these, only some of the document is marked as plagiarised, probably revealing something about EVE's string search strategy. However EVE did find multiple references, four sites for Babbage, including the original and six sites for Gosling, four of which are mirrors of the white paper and two of which quote sections from it.

EVE is probably useful for the isolated cases where plagiarism is suspected but the source cannot be traced, but, in general, one of the Web based services would seem to be more effective.

4. COMPARATIVE EVALUATION

Whilst the results returned by the plagiarism detection services are broadly comparable, they represent different ends of the spectrum for tutors in terms of ease-of-use and effectiveness. There are also security and operational issues to be considered.

From an HCI perspective, the administrative interfaces to plagiarism.org, copycatch.com and Integriguard are very similar. Each uses a standard navigation bar to the left of the Web browser window, with related information displayed on the right. The copycatch.com interface is arguably the best of the three, allowing tutors to easily register themselves and establish assignment cut-off dates. The plagiarism.org interface suffers from being cluttered making it can be difficult to find the required section of the site. The Integriguard interface seems more geared towards advertising its service than aiding instructors to use it.

All three services work along the same principles. A tutor registers a submission, in some cases with a cut off date. Students then register on a designated student section of the site and submit their paper. The results are then sent to the tutor. In the case of Integriguard and plagiarism.org, processing can be done at any time after the paper is received. For copycatch.com, all submissions are batch processed after the due date. The source code detection service, MOSS, uses a similar batch-processing process, only in this case submissions

are collected locally and bulk submitted to MOSS. This means that late submissions can be given the same amount of automated scrutiny as submissions received on time.

The submission process for students ranges from poor to satisfactory. Plagiarism.org and Integriguard provide only a simple text box for a student to cut and copy their text into. Copycatch.com allows students to select a file from their local hard drive in popular word processing formats and submit that instead. None of the methods are geared towards helping students unfamiliar with computer operation and tutors may find themselves having to provide support for the submission process.

Aside from this, the workload for tutors using the services is low. They merely need to set up the service to accept papers, ensure that all students expected to submit assignments do so and interpret the results at the end. The Web-based results from plagiarism.org and copycatch.com are much more readable than the e-mail returned by Integriguard. For security purposes, viewing the results from copycatch.com requires knowledge of the instructor's password, viewing those from plagiarism.org (or MOSS for that matter) does not.

From an overall effectiveness viewpoint, the results from copycatch.com, plagiarism.org and MOSS are all comprehensive and well presented. Integriguard suffers since only a fraction of the essay is ever checked for Web plagiarism. The same is true for the EVE tool, although it is possible to increase the number of tests at the expense of a much longer processing time. For free-text plagiarism detection, there is little to choose between plagiarism.org and copycatch.com; both seem effective at finding suspected plagiarism and communicating this to tutors, although copycatch.com is presently free, as is MOSS, for source code.

Since the services are all based in the US, there might be implications with regard to the Data Protection Act. These are not considered within the context of this paper.

5. CONCLUSIONS

The area of free-text anti-plagiarism is still in its infancy and will almost certainly continue to receive plenty of media attention as more incidents arise. Some universities do not yet appreciate the full scope of the problem, we encourage them to operate a pro-active anti-plagiarism policy and to take steps to actively seek out plagiarism in student submissions.

We, the authors, are continuing to research and develop software to detect free-text plagiarised submissions. The existing Web based systems are able to trace some simple cases of plagiarism, but we believe that more sophisticated linguistic metrics could be more effective. Our initial investigations are

proving successful and suggesting that more powerful detection systems are possible. The existing services are costly (integriguard.com, plagiarism.org) or may start to charge at any time (copycatch.com, MOSS). As a result we foresee the need for a plagiarism Web server based in the UK available for free academic use.

One problem that we have foreseen, is that it is currently difficult to discuss plagiarism, since the term is ill-defined and the language limited. We have prepared a descriptive taxonomy of plagiarism that aims to aid discussions in the area [1]. Further work will aim to encourage departments to implement a pro-active anti-plagiarism policy, where plagiarism is not only taken seriously when found, but is actively sought out during all coursework activities.

Finally, whilst there seems to be little to choose between the services on offer at present, we would encourage all academic institutions to show to re-evaluate their plagiarism policies and seriously consider using one of these services as a matter of course. In particular the MOSS service is an asset to the academic computing community. Whilst it might charge in the future, it is currently free for academic use and deserves to be more widely used.

6. REFERENCES

- [1] Culwin F. & Lancaster T., *A Descriptive Taxonomy of Plagiarism*. Awaiting publication, available from South Bank University, London (2000).
- [2] Culwin F. & Naylor J., *Pragmatic Anti-Plagiarism*. Proceedings Third Conference on the Teaching of Computing, DCU Dublin IE (1995).
- [3] Denhart A., *The Web's Plagiarism Police*. Available at <http://www.salon.com/tech/feature/1999/06/14/plagiarism/index.html> (1999).
- [4] Gajadhar J., *Issues in Plagiarism for the New Millennium: An Assessment Odyssey*. Available at http://www.asee.org/prism/december/html/student_plagiarism_in_an_onlin.htm (1998).
- [5] Verco K. & Wise M., *Software for Detecting Suspected Plagiarism: Comparing Structure & Attribute Counting Systems*. Proceedings First Australian Conference on Computer Science Education, Sydney, Australia (1996).
- [6] Wise M., *YAP3: Improving Detection of Similarities in Computer Program and Other Texts*. Proceedings ACM SIGCSE '96, Philadelphia, USA (1996).
- [7] *Copycatch.com*. Available at <http://www.copycatch.com>.
- [8] *Essay Verification Engine*. Available at <http://www.canexus.com/eve/>.
- [9] *Integriguard*. Available at <http://www.integriguard.com>.
- [10] *Measure of Software Similarity*. Available at <http://www.cs.berkeley.edu/~moss/general/moss.html>.
- [11] *Plagiarism.org*. Available at <http://www.plagiarism.org>.
- [12] *Year 2000 Computer Problem (Y2K Bug)*. Available at <http://www.cheater.com/homework/Homework/Schoolsucks/Uploads/Random2/10791413.htm>.
- [13] Babbage C., *Passages from the life of a Philosopher, Chapter VIII*. Available at <http://www.fourmilab.ch/babbage/lpae.html>.
- [14] Gosling J. & McGilton H., *The Java Language Environment – A White Paper*. Available at <http://www.javasoft.com/docs/white/langenv/index.html>.