Design and Evaluation of Plagiarism Prevention and Detection Techniques in Engineering Education

ARTICLE HISTORY
Compiled January 3, 2019

ABSTRACT
Higher education students are expected to develop critical analysis and creative thinking skills, where plagiarism can damage the development of these skills in addition to damaging the whole education process and experience. Furthermore, plagiarism undermines the trust between the lecturers and students and the reputation of the academic institutions can be affected if plagiarism is not considered seriously, where the degrees offered by these institutions can be devalued. In this paper, two plagiarism prevention techniques followed by two plagiarism detection techniques used in the engineering education in the University of Southampton are presented. The plagiarism prevention techniques presented are based on assigning individual coursework specifications to students and the use of individual presentation of coursework findings. Then, the plagiarism detection techniques are based on detecting the writing styles of students and testing the student' codes in different configurations.

KEYWORDS
Plagiarism detection; plagiarism prevention; engineering education

1. Introduction

In the age of information technology and the wide use of laptops, tablets and smart phones by students, it has become easy for students to access and use information online. Hence, plagiarism detection and prevention has become a major concern is higher education worldwide. According to the University of Southampton handbook, “Plagiarism is using someone else’s work without acknowledging it or crediting the original author” (University of Southampton 2017b).

Plagiarism has a number of negative effects on education. Students who plagiarise lose the chance to develop their critical thinking and research experience they are supposed to gain in higher education. Additionally, plagiarism generally affects the relations between the lecturers and students, where trust can be lost. Furthermore, the reputation of academic institutions can be destroyed and their degrees devalued, if plagiarism is allowed to be the norm (Dey and Sobhan 2006). Therefore, universities often educate students about plagiarism, apply plagiarism prevention and detection techniques and then use appropriate punishments in order to deter students. The University of Southampton takes plagiarism very seriously and applies a range of penalties to deter students, which can range from failing an assignment to failing a class or getting suspended or expelled (University of Southampton 2017b,a).

There have been several techniques for detecting plagiarism, which mainly use software tools for detecting the different types of plagiarism (Halak and El-Hajjar 2016; Li et al. 2004; Jhi et al. 2015; Tian et al. 2015; Rosales et al. 2008a; Cosma and Joy 2012; M. Jiffriya, M. Jahan, R. Ragel and S. Deegalla 2013). These techniques are mostly based on comparing text and finding any textual similarity to published material available in the software repositories, which is then validated by an instructor (Li et al. 2004). The "Turnitin" software tool (Turnitin 2017) is an example of such a tool, which is widely adopted by the academic institutions in the UK including the University of Southampton.

While using software tools for detecting plagiarism can be effective in highlighting potential plagiarism cases, these tend to be mainly textual plagiarism, where the text is directly copied from its source without paraphrasing or acknowledging the source. The software tools tend to be less effective to highlight ideas-theft or collusion or software plagiarism. This is due to the fact that these software tools compare the text of a report with the text in submissions available in the repository, which makes these tools not efficient in detecting the plagiarism of undocumented ideas. On the other hand, the repository of these software tools might be limited in covering all relevant literature and hence they may fail to detect plagiarised material (Kaner and Fiedler 2008a).

Another common case can be found in class-based assignments, where students share the solution to the common problems. This is particularly relevant in engineering coursework assignments, where students are requested to develop a hardware or software using specific configurations. Source code plagiarism detections (Rosales et al. 2008a; Cosma and Joy 2012) have been developed to detect such cases, but students can use different programming languages to implement the same solutions, where detection of plagiarism using these tools becomes inefficient.

In this paper, we commence by presenting a detailed description of plagiarism and its types in Section 2. Then, we present two techniques for preventing plagiarism in Section 3 and another two techniques to detect plagiarism in engineering class assignments in Section 4. The plagiarism prevention techniques presented are based on assigning individual coursework specifications to students and the use of individual presentation of coursework findings. On the other hand, the plagiarism detection techniques are based on evaluation of the writing style of technical reports and the rigorous testing of the software codes submitted by students. These techniques are currently applied to courses at the undergraduate and Master level in the University of Southampton, where we show that they are effective at reducing plagiarism and improving students' understanding. Finally, we present the conclusions in Section 6.

2. Plagiarism definition and types

According to Merriam-Webster online dictionary, to "Plagiarize" means "to use the words or ideas of another person as if they were your own words or ideas". In their article "What is Plagiarism" (iParadigms 2017), iParadigm list six points that explain the different forms of plagiarism as follows (iParadigms 2017):

(1) "turning in someone else's work as your own;
(2) copying words or ideas from someone else without giving credit;
(3) failing to put a quotation in quotation marks;
(4) giving incorrect information about the source of a quotation;
(5) changing words but copying the sentence structure of a source without giving

credit;

(6) copying so many words or ideas from a source that it makes up the majority of your work, whether you give credit or not".

Plagiarism includes submitting someone else's work as your own, copying words or ideas from someone else without giving credit or without using quotation marks (A.M. El Tahir-Ali, H.A. Dahwa-Abdulla and V Snasel 2011). Additionally, changing the words in a copied sentence without giving credit to the source and using words and ideas from a source such that most of the work is not your own are considered to be plagiarism (A.M. El Tahir-Ali, H.A. Dahwa-Abdulla and V Snasel 2011). Therefore, plagiarism has been categorized as text plagiarism, style plagiarism and ideas plagiarism (T. Elfadil, E. Naomie, and S.S. Alzahrani 2015; A.M. El Tahir-Ali, H.A. Dahwa-Abdulla and V Snasel 2011). The text plagiarism can be to copy and paste the exact sentences or phrases without using quotations or proper citations. It can also be to take sentences from their source and change the order of words, again without using quotations or proper citations. On the other hand, style plagiarism consists of copying others' style of reasoning and writing, even if the text is paraphrased, while not using proper citations. Finally, idea plagiarism consists of claiming others' ideas as one's own ideas (A.M. El Tahir-Ali, H.A. Dahwa-Abdulla and V Snasel 2011). Plagiarism of ideas is the most difficult to track (Graduate School and University Center, City University of New York 2012), since the same idea can be rewritten and explained in a different way than the original work, while not changing the idea.

In addition to the moral issues associated with claiming other's work, plagiarism has several academic and legal implications (Cully 2013). From an academic perspective, the quality of academic degrees stems from the fact that those receiving the academic qualification must have a predetermined level of input and contribution. These contributions are required so that the students can show they have learned and can use the knowledge gained to produce some output. Hence, plagiarism would destroy the quality of any qualification, if allowed to become the norm. Additionally, students would lose the opportunity to learn the required information that would prepare them to be successful in their careers (Cully 2013).

Most scientific developments are built on previous discoveries, which are considered to be prior knowledge (Graduate School and University Center, City University of New York 2012). Hence, all scientific papers cite the prior knowledge work in order to acknowledge the prior work as well as give more credibility for the produced work. Therefore, it is essential that students in higher education are educated about plagiarism and its effect on their education experience as well as their professional experience later. Hence, several plagiarism detection and prevention techniques have been employed in higher education for supporting the students' learning experience, as described in the following sections.

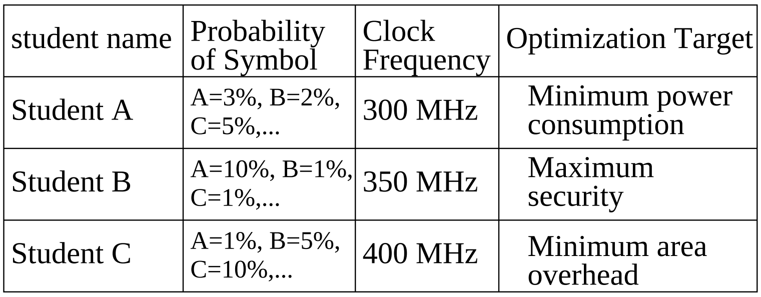
3. Plagiarism prevention techniques

In this section, we describe two plagiarism prevention techniques we use in our department and we believe are powerful in reducing the plagiarism in assignments.

3.1. *Assigning individual coursework specifications*

This technique can be applied in lab-based engineering coursework, where each student is typically required to develop a piece of hardware or software based on pre-defined requirements such as functionality, performance and so on.

In this case, it is natural for students to discuss the problems in hand and exchange ideas on how to meet the specifications. This form of collaboration is actively encouraged in the engineering domain, as it enhances team working, which is a vital skill for a successful engineer. However, some students may take advantage of such environment and plagiarize solutions from their classmates, which is called collusion. To prevent such practices, in this section we suggest to assign a unique specification for each student, such that each student will have to develop his/her own design and are no longer able to copy assignments from their colleagues. We designed this method such that it does not increase the amount of assessment a teacher will need to do, otherwise it may become impractical. To illustrate further, in the following we provide two examples to show how this method can be applied. The first is a digital design assignment, where each student is required to develop a Huffman-based compression circuit. In this exercise, each student is given a unique functional specification by changing the probability of the characters in the data stream to be compressed and also each student has a distinct optimization target as shown in Figure 1, which shows an illustrative set of specifications. Using this method, students can discuss the methods to approach this design problem, but each student will ultimately need to develop his/her own circuit. To mark such assignments the teacher may need to spend more time checking the individual design for each students, but such a small increase in the marking effort is significantly outweighed by the reduction of plagiarism cases that would have taken place otherwise.



student name	Probability of Symbol	Clock Frequency	Optimization Target
Student A	A=3%, B=2%, C=5%,...	300 MHz	Minimum power consumption
Student B	A=10%, B=1%, C=1%,...	350 MHz	Maximum security
Student C	A=1%, B=5%, C=10%,...	400 MHz	Minimum area overhead

Figure 1. An exemplar unique specification for a Digital Design assignment.

The second example is a Cryptography coursework, where each student is given a unique cipher in order to perform cryptanalysis with the aim of cracking it. In this case, students can discuss different approaches of cryptanalysis but they have to decipher their own unique message.

In order to evaluate how effective the approach of assigning individual specifications to different students in reducing the collusion-related plagiarism, we used the similarity score between reports submitted to the same assignment, which is compared in two subsequent years. We considered two course, the first is called system-on-chip design, a compulsory module for one of our master programs and the second is Cryptography, which is a fourth year module given to a wide range of programmes in both Computer Science and Electronics Engineering. In both cases, we have observed a significant decrease in the inter-report similarity scores obtained from Turnitin software. For the digital design assignment, the maximum inter-similarity score reduced by 16% and the number of similarity cases reduced to the half. Similar trend has also been observed for the Cryptography module with a reduction of 13% of the similarity metric obtained by the Turnitin software.

3.2. *Using individual presentation*

In the previous section, we described how assigning individual coursework specifications can help to reduce the collusion-relate plagiarism. However, this method might not be effective in detecting plagiarism of undocumented ideas, for example when someone other than the student does their assignment for them. Therefore, we use "individual presentation technique", where we ask students to present and explain their results in front of the whole class or only to the lecturer. This technique can be used to detect plagiarism of undocumented ideas in the commonly-used 'group design projects' in engineering education. In this type of assignments, the class is divided into groups, where each group is required to design a system using hardware or software tools. In group projects in general, some students tend to contribute more than others to the design and development of the projects. Therefore, we use individual presentations, where we ask students to explain their own contribution. This can help reduce cases of plagiarism, where each student can only claim credit for his/her own contribution to the project.

We have opted to use the distribution of marks in the class in order to estimate the effectiveness of this technique. The distribution of the class marks tends to be normal distribution for large sized cohorts (PS.L. Hoskins, and J.C van Hooff 2005), which reflects of the variable skills and abilities of students. For the purpose of illustrating the effectiveness of using individual presentation in reducing plagiarism, we considered the marks' distribution for a coursework given to Master students in the University of Southampton, where the coursework contributes 10% of the final mark. Figures 2 and 3 show the distribution of marks for similarly sized cohorts before and after using the individual presentation, respectively.

In Figure 2, we show the distribution of marks for a class of 48 students, where these marks were obtained before using the individual presentation. Then, in the following academic year, we told students that we will select at least 10% of the students at random to present their results. Then, we asked a sample of students to present their results after the submission of assignments. Figure 3 shows the distribution of marks after applying the individual presentation techniques, where the figure shows that the marks have a more normal distribution than that in figure 2. Before using the individual presentation technique, the average mark was 8.1 out of 10 with a small standard deviation of 0.9. On the other hand, after we used the individual presentation technique, the average mark dropped to 6.1 out of 10 with a larger standard deviation of 1.4. Therefore, the use of individual presentation might have affected the students' views of plagiarism, where they tried to avoid plagiarising ideas or solutions, which is shown in the more normal distribution of the marks after applying this technique (PS.L. Hoskins, and J.C van Hooff 2005).

4. Plagiarism detection techniques

University students are usually educated about plagiarism, where in the University of Southampton, we teach our students in their first year to avoid plagiarism and we

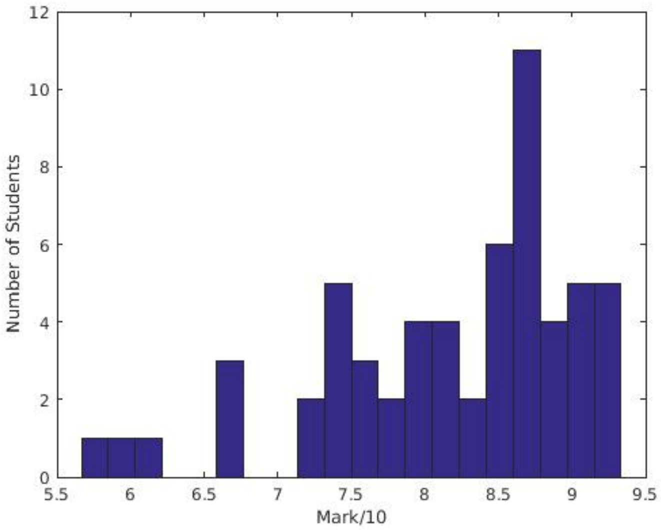


Figure 2. Distribution of students' marks before the use of individual presentation method.

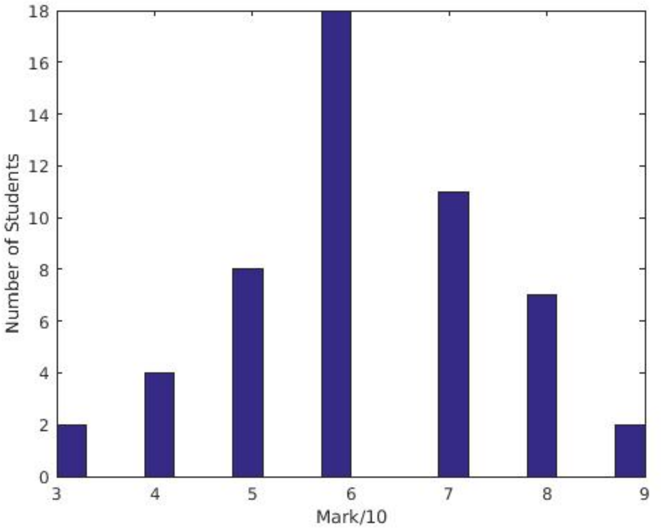


Figure 3. Distribution of students' marks after the use of individual presentation method.

educate them about the ways to avoid plagiarism (University of Southampton 2017b). We teach them to avoid any form of plagiarism by extensive paraphrasing coupled with appropriate referencing using reliable and relevant literature sources (Cully 2013). Additionally, as the aim of their work is to assess their understanding, students learn that the best way to avoid plagiarism and show their understanding is to present and describe their work using their own words. Throughout their education, students have to present several assignments and projects, where they would present generally known facts and information found in many works. Hence, we teach students to avoid plagiarism by avoiding the "copy and paste" and by writing their own understanding of the information using their own words. Additionally, we teach students to use proper referencing of reliable sources and we also teach them the referencing styles they can use (University of Southampton 2017b; Cully 2013).

There are many online software tools aimed at detecting plagiarism of which we use Turnitin (http://turnitin.com) in the University of Southampton. Turnitin has a database that contains archives of all previously submitted work as well as access to papers, reports and books available in the Internet. A technical review of plagiarism detection systems was reported in (J. Bull, C. Collins, E. Coughlin, and D. Sharp 2001; Chester 2001), where recommendations were made to the Joint Information Systems Committee (JISC) in the United Kingdom (UK). Afterwards, Turnitin was recommended by JISC as the online commercial detection tool to be used by all higher education institutions in the UK. The similarity detection algorithms for Turnitin is a commercial secret, however according to (H. Maurer, F. Kappe, and B. Zaka 2006) the most commonly used techniques in document comparison software involve word fingerprinting, where strings from a document, referred to as fingerprints, are compared for similarities with preprocessed indexes from other documents.

However, it has been shown in (B. Marsh 2004; Weber-Wulff 2008) that Turnitin is not able to handle paraphrased texts effectively. Therefore, a combination of human checking and software checking might be necessary for plagiarism detection. The similarity report produced by Turnitin should be checked by academics in order to make a fair decision on the plagiarism detection of students (Kaner and Fiedler 2008b). This is due to the fact that Turnitin compares all text included in the submitted documents and it shows technical phrases or the references as copied material, when they are not. Therefore, we normally would study the similarity report produced by Turnitin in order to detect plagiarism and we do not depend only on the similarity index produced by the Turnitin software.

Therefore, in addition to using Turnitin we have devised two techniques that we use to detect plagiarism. In the following we will present these techniques, which are based on the writing style of students and the rigorous testing of the software codes submitted by students.

4.1. *Using writing style*

The first plagiarism detection technique used by university academics is the writing style difference between students' own work and the work copied from literature (A.M. El Tahir-Ali, H.A. Dahwa-Abdulla and V Snasel 2011; T. Elfadil, E. Naomie, and S.S. Alzahrani 2015; Graduate School and University Center, City University of New York 2012). As academics, we know that published material normally goes through several iterations of revision in order to improve the language, the flow of information and organization of the work. Some students tend to overlook this fact and simply "copy and paste" the published material in their work, where academics can easily spot the difference between the two writing styles and hence detect plagiarism. Additionally, to prove the plagiarism detection, academics can use an Internet search to find the source used by students by simply searching for phrases used in the students' submitted work (T. Elfadil, E. Naomie, and S.S. Alzahrani 2015; Graduate School and University Center, City University of New York 2012).

Furthermore, academics can detect plagiarism in students' work by considering the quality of images and tables in their submitted work. Copied figures, tables and equations tend to have lower quality than those produced by students, when printed on paper, and these can have different style than other figures and tables in the same report. Hence, plagiarism of figures, tables and equation can be detected. Therefore, we always teach our students to produce their own figures, tables and equations and always cite the work they used to learn about these.

The following is an example taken from a student's report, where it is obvious that there are more than one writing style. In the background section of the report, the student includes the following description:
"A simple spatial diversity technique, which does not involve any loss of bandwidth, is constituted by the employment of multiple antennas at the receiver, where several techniques can be employed for combining the independently fading signal replicas, including Maximum Ratio Combining (MRC), Equal Gain Combining (EGC) and Selection Combining (SC). Several transmit - rather than receive - diversity techniques have also been proposed in the literature [2], [3], [4], [5]."

While reading this paragraph, we would not suspect it as plagiarized as it could be written by a good student with good understanding and good writing skills. However, when we read the paragraph written immediately after this one, we can suspect a plagiarism case. The paragraph written after the above quotation is the following:
"Multiple antenna systems have the potential to achieve much higher bandwidth efficiencies than single antenna systems in fading environmental. [1] In vertical Bell-labs Layered space-time (VBLAST), each layer is independently and associated with a certain transmit antenna. Treating a VBLAST system as a multi-user system enables interference suppression and successive interference cancellation (SIC) to be used in detection. [6] For example, a zero forcing (ZF) SIC algorithm with optimum ordering (ZF-VBLAST) and combined a minimum mean square error (MMS?-VBLAST). However, both algorithms involve the computational of the pseudo-inverse of a matrix, has cubic complexity. Although the performance is degraded, the computational effort at the receiver is reduced enormously. However, through researches showed that the performance of ZF-VBLAST becomes bad while the computational complexity is much lower.[2]".

After reading the whole section of the above report, it has become clear that there are more than one writing styles in the section, as indicated in the two quotation above. What we did in this case is simple copy a sentence from the "better style" quotation and do an Internet search, where it would be easy to identify the sources of these sentences. To illustrate further, the first quotation is very well written, while the second quotation has several structural and grammatical mistakes such as using the reference number after the full stop. Some other examples showing the mistakes in the second quotation are the following phrases "each layer is independently and associated" and "For example, a zero forcing (ZF) SIC algorithm with optimum ordering (ZF-VBLAST) and combined a minimum mean square error (MMS?-VBLAST).", which are wrong and non-complete English sentences.

4.2. *Testing software submitted by students*

It is a common practice in engineering education that students are given assignments to design a system and simulate it in software. Hence, we first make sure that we change the assignment or project every year, so that students do not have the chance to take the designs and codes from students in previous years. This is a plagiarism prevention technique we use, as described in the previous section.

For every assignment or project, we ask students to submit their reports as well as their software code. In addition to checking their similarity report produced by Turnitin, we would test the students' code in the configurations given to them in order to make sure it produces the required output. The project configurations can include the number of errors, run time and functionality for example (Rosales et al. 2008b). In addition to the configurations given to students in the assignment instructions, we test the code using other configurations in order to make sure the code works as it is supposed to and that students did not simply build the system to work in the configurations set in the assignment. In order to save time in testing all students' code, we built a script that would run all students' code automatically and test them in all possible configurations. The script will then produce a report for each code showing the output for all configurations and which configurations did not give the right output. The report also includes the time it took the code to complete in each configuration and how well the code is commented and how long the code is.

We can detect plagiarism by comparing the report produced for each code and more precisely by considering the run time of the code, the length of the code and the commenting style. In engineering, we expect different people to have different coding style and different logic and hence if we find two identical reports, then we would suspect a plagiarism case and then we will examine the codes for checking if there is any plagiarism and we also would study the similarity reports produced by Turnitin.

5. Evaluation

In order to evaluate the existing practices in the University of Southampton for the prevention and detection of plagiarism, we have conducted a survey, which includes five questions as shown in Table 1. We had 10 participants in this survey, which included two professors, four associate professors and four lecturers.

For question 1, 80% of the participants reported a low level of plagiarism in their courses and the remaining 20% indicated a moderate level of plagiarism. For questions 2, the results showed that 100% of the participants use software packages such as Turn-it-in to detect plagiarism and 60% use the writing style. Additionally, 20% of the participants indicated that they also carry interviews with the students if they suspected a plagiarism has been committed.

Responses to question 3 showed that 100% of the participants are willing to consider using other plagiarism detection methods. This indicates a willingness from academics to consider new approaches to improve the effectiveness of the current methods. More explicitly, some participants reported the need for automated tools capable of detecting plagiarised codes and this is a particularly challenging problem in the engineering field.

Responses to question 4 showed that 60% of participants rely on the use of the individual presentation technique to prevent plagiarism, while 60% use the individual coursework specification method. In addition, 20% of the participants indicated that they have dedicate a lecture in their courses to explain the meaning of plagiarism and how to avoid it. According to the survey, this additional training has helped increase the awareness of the students and reduce the number of plagiarism cases. Finally, responses to question 5 have shown that the participants are happy with their chosen approaches and they do not think there is a need to adopt other techniques.

Overall, the survey results have shown that the proposed plagiarism detection methods are being adopted and are perceived to be effective. On the other hand, the participants felt more techniques to detect the plagiarisms of codes can enhance the existing practices.

Table 1. Survey questions.

Please answer all the questions below, you can choose more than one answer for the multiple choice questions

1. Do you think the plagiarism level in your courses is
1) Low
2) Moderate
3) Significant

2. Which of the following plagiarism detection techniques you currently use?
1. Writing Style
2. Turn in in or other plagiarism detection software
3. Other techniques (please specify)

3. Would you consider using other techniques for plagiarism detection?
Yes/No

4. Which of the following plagiarism prevention technique you currently use:
1. Individual Presentation
2. Individual coursework specifications
3. Other techniques (please specify)

5. Would you consider using other techniques for plagiarism prevention?
Yes/No

6. Conclusion

In this paper, we presented a definition of plagiarism and its categories followed by plagiarism prevention and detection techniques we use in our engineering education programmes at the University of Southampton. We presented plagiarism prevention techniques based on assigning individual coursework specifications and using individual presentation. Then, we explained two plagiarism detection techniques based on detecting the writing styles of students and testing the students' codes in different configurations. Furthermore, we have provided examples from our undergraduate and Master courses offered in the University of Southampton in order to show the effectiveness of these techniques in preventing and detection plagiarism.

References

A.M. El Tahir-Ali, H.A. Dahwa-Abdulla and V Snasel. 2011. "Overview and comparison of plagiarism detection tools." In *Dateso 2011: Annual International Workshop on DAtabases, TExts, Specifications and Objects*, April, 1–10.

B. Marsh. 2004. "Turnitin.com and the scriptural enterprise of plagiarism detection." *Computers and Composition* 21 (4): 427–438.

Chester, G. 2001. *Pilot of free-text electronic plagiarism detection software.* Technical Report. Joint Information Systems Committee (JISC), London, UK.

Cosma, G., and M. Joy. 2012. "An Approach to Source-Code Plagiarism Detection and Investigation Using Latent Semantic Analysis." *IEEE Transactions on Computers* 61 (3): 379–394.

Cully, P. 2013. “Plagiarism avoidance in academic submissions.” *Dublin Institute of Technology* .

Dey, S. K., and M. A. Sobhan. 2006. “Impact of Unethical Practices of Plagiarism on Learning, Teaching and Research in Higher Education: Some Combating Strategies.” In *2006 7th International Conference on Information Technology Based Higher Education and Training*, July, 388–393.

Graduate School and University Center, City University of New York. 2012. “Avoiding and Detecting Plagiarism: A Guide for Graduate Students and Faculty.” *Graduate School and University Center, City University of New York* .

H. Maurer, F. Kappe, and B. Zaka. 2006. “Plagiarism - a survey.” *Journal of Universal Computer Science* 12 (8): 1050–1084.

Halak, B., and M. El-Hajjar. 2016. “Plagiarism detection and prevention techniques in engineering education.” In *2016 11th European Workshop on Microelectronics Education (EWME)*, May, 1–3.

iParadigms. 2017. “iParadigms: what is plagiarism?” http://www.plagiarism.org/plagiarism-101/what-is-plagiarism/.

J. Bull, C. Collins, E. Coughlin, and D. Sharp. 2001. *Technical review of plagiarism detection software report.* Technical Report. University of Luton, London, UK.

Jhi, Y. C., X. Jia, X. Wang, S. Zhu, P. Liu, and D. Wu. 2015. “Program Characterization Using Runtime Values and Its Application to Software Plagiarism Detection.” *IEEE Transactions on Software Engineering* 41 (9): 925–943.

Kaner, C., and R. L. Fiedler. 2008a. “A Cautionary Note on Checking Software Engineering Papers for Plagiarism.” *IEEE Transactions on Education* 51 (2): 184–188.

Kaner, C., and R. L. Fiedler. 2008b. “A Cautionary Note on Checking Software Engineering Papers for Plagiarism.” *IEEE Transactions on Education* 51 (2): 184–188.

Li, Ming, Xin Chen, Xin Li, Bin Ma, and P. M. B. Vitanyi. 2004. “The similarity metric.” *IEEE Transactions on Information Theory* 50 (12): 3250–3264.

M. Jiffriya, M. Jahan, R. Ragel and S. Deegalla. 2013. “AntiPlag: Plagiarism detection on electronic submissions of text based assignments.” In *8th IEEE International Conference on Industrial and Information Systems*, December, 376–380.

PS.L. Hoskins, and J.C van Hooff. 2005. “Motivation and ability: Which students use online learning and what influence does it have on their achievement?” *British Journal of Educational Technology* 36 (2): 177–192.

Rosales, F., A. Garcia, S. Rodriguez, J. L. Pedraza, R. Mendez, and M. M. Nieto. 2008a. “Detection of Plagiarism in Programming Assignments.” *IEEE Transactions on Education* 51 (2): 174–183.

Rosales, F., A. Garcia, S. Rodriguez, J. L. Pedraza, R. Mendez, and M. M. Nieto. 2008b. “Detection of Plagiarism in Programming Assignments.” *IEEE Transactions on Education* 51 (2): 174–183.

T. Elfadil, E. Naomie, and S.S. Alzahrani. 2015. “Existing plagiarism detection techniques.” *Online Information Review* 39 (3): 383–400.

Tian, Z., Q. Zheng, T. Liu, M. Fan, E. Zhuang, and Z. Yang. 2015. “Software Plagiarism Detection with Birthmarks Based on Dynamic Key Instruction Sequences.” *IEEE Transactions on Software Engineering* 41 (12): 1217–1235.

Turnitin. 2017. “Turnitin.” http://turnitin.com/.

University of Southampton. 2017a. “University of Southampton Calendar: Plagiarism and Cheating: Policy and Procedures.” http://www.calendar.soton.ac.uk/sectionIV/sectIV-index.html.

University of Southampton. 2017b. “University of Southampton Plagiarism Policy: Student Handbook.” http://www.southampton.ac.uk/studentservices/handbook/index.html.

Weber-Wulff, D. 2008. “On the utility of plagiarism detection software.” In *3rd International Plagiarism Conference, Newcastle upon Tyne, UK*, 1–11.