Plagiarism Detection Software Test 2013

Debora Weber-Wulff[1], Christopher Möller[1], Jannis Touras[2], Elin Zincke[2]
[1] HTW Berlin, [2] HU Berlin

Abstract

Can software automatically detect plagiarism? Many companies sell software that suggests just that. Prof. Dr. Debora Weber-Wulff, professor for media and computing at the HTW Berlin, has previously conducted six tests of plagiarism detection systems, in 2004, 2007, 2008, 2010, 2011, and 2012. For 2013, instead of attempting to test all possible systems, a selection was made that included software previously found to be at least partially useful, as well as some newcomers. In all, 28 systems were investigated, but only 15 systems were able to complete the test series that included many new test cases designed to address specific aspects of the use of plagiarism detection systems at educational institutions. In particular, large files that simulated bachelor's and master's theses were constructed, one test case was designed to determine if the software can access and use Google Books, and some test cases that use cheats sometimes used by students to thwart such software were put together. In addition, Hebrew was used as the non-Latin test case language 2013.

The results are comparable with previous years: Even if some of the systems are easier to use now, they still do not produce the documentation that would be necessary in Germany for presentation to an examination board. Most troublesome is the continued presence of false negatives – the software misses plagiarism that is present – and above all false positives. When systems report significant plagiarism for common phrases, or even for a paper that is completely original, using these results without close examination may cause grave damage. In particular, the numbers reported by the systems are not consistent and should be treated only as possible indicators, not as absolute judgment values.

0. Introduction

Since 2004 researchers at the HTW Berlin have been testing so-called plagiarism detection software in order to determine how effective it is and how well it fits into university use cases. The HTW Berlin has kindly financed the student researchers and the equipment necessary for these tests so that they can be conducted as independently of the software producers as possible. We were given free access to each of the software systems for which the tests could be completed, but this of course means that the software producers know exactly what it is that we were testing.

There have been six tests conducted up until now: 2004, 2007, 2008, 2010, 2011, and 2012. The first four tests were general tests of plagiarism detection software. In 2011 the thesis of former German defense minister Karl-Theodor zu Guttenberg was tested with the top five systems from the 2010 test. A specialized test of collusion detection systems, ones that detect copying within a closed group of documents, was conducted in 2012.

The results are available at the Portal Plagiarism hosted at the HTW Berlin http://plagiat.htw-berlin.de/software and there have been a number of publications about the results (Weber-Wulff & Wohnsdorf 2006; Weber-Wulff & Pomerenke 2007; Weber-Wulff 2008; Weber-Wulff 2009; Weber-Wulff & Köhler 2011a; Weber-Wulff & Köhler 2011b; Portal Plagiat).

Although the results of past years have always demonstrated that such software is not a solution to the problem of plagiarism, most particularly because of constant problems with false positives and false negatives, as well as massive usability problems, many universities still want to purchase such software. There has been much discussion, especially in Germany, about plagiarism in the aftermath of the zu Guttenberg plagiarism scandal (GuttenPlag Wiki 2011) and the continuing documentation of major plagiarism in dissertations throughout Germany (VroniPlag Wiki 2013). Educators are quite concerned, in particular because they are more often being confronted with text taken from the Internet – and quite often from the Wikipedia – in texts that their students submit for grading. They wish, understandably, for some sort of litmus test that will weed out the plagiarisms before they have to embark on the ordeal of reading and grading the flood of papers.

As it turns out, the plagiarism problem is an extremely complex one. The first author will be addressing many facets of this in a forthcoming book (Weber-Wulff 2014). There are many questions that present themselves when looking at a possible mechanical determination of plagiarism:

- What exactly constitutes plagiarism? Just copy & paste, or paraphrasing without sourcing, or taking ideas?
- How much of a paper can be copied without it being considered plagiarism?
- Is it only plagiarism if it was copied on purpose?
- Do the systems count the number of characters, of words, of sentences?
- How is the amount of plagiarism quantified?
- Do the systems check the entire text, or just a sample?

Educators want something simple to use and reliable in its results – getting a different result 10 minutes later is not a good option. However, it is not possible to leave the decision of whether or not something is a plagiarism to a machine – it is vital for a human being to take that final decision. Plagiarism detection software is only a tool, not an infallible test.

In this report the methodology of the current test will first be discussed, including the make-up of the test cases and the criteria used for evaluation. Section two will give more detail about the test cases used. The third section will give a summary of the results. A few recommendations for universities and software companies are given in section four. The individual results of the tests and the systems not tested are given in an additional document as an appendix.

1. Methodology

In this section, the motivation for the choice of systems will be discussed, as well as an overview of the test cases and the evaluation criteria. A brief description of the procedures used for conducting the tests will then be given.

1.1 Choice of Systems

In previous tests, efforts had been made to test all more or less serious offers of plagiarism detection services, although many turned out to be untestable. It was decided for the test 2013 to begin with systems that had either been determined to be at least partially useful in past tests, or which had announced a new version of their system. In addition, a few new systems that had never been tested were also included in the lineup.

The following 15 systems were able to complete the entire test:

> Compilatio, Copyscape, Docoloc, Duplichecker, Ephorus, OAPS, PlagAware, Plagiarisma, PlagiarismDetect, PlagiarismFinder, PlagScan, PlagTracker, Strike Plagiarism, Turnitin, Urkund

An attempt was also made to test the Hungarian system KOPI, as it was being advertised as able to find plagiarism by translation. As it turned out, it could only deal with translations from the English-language Wikipedia into Hungarian, produced unintelligible reports and tended to revert to Hungarian at times; the test had to be discontinued.

There were 13 other systems that were investigated but could not be tested for a variety of reasons. Some are just different product names for the same systems (iThenticate and WriteCheck use the same database as Turnitin), or because they also offer editing and ghostwriting services, or because we were unable to obtain access to the system. We do not consider a company that offers plagiarism detection and ghostwriting or anti-plagiarism-detection-system services to be viable for serious use at an education institution. The following twelve systems were not tested:

> Academic Plagiarism, AntiPlag, Custom Writings, Effective Papers, iThenticate, PaperRater, The Pensters, Plagiarism Checker, Plagium, PlagSpotter, Small SEO Tools, WriteCheck

A summary of the test for each individual system and the reasons for not testing the other systems can be found in the appendix in a separate file, and the individual scores of the systems are recorded online at the Portal Plagiarism.

1.2 Test case overview

Past tests that were conducted at the HTW Berlin have always used hand-made test cases that are assumed to mirror typical student plagiarisms. The cases are, indeed, rather small – only one to two pages – as they are also used as exercises for the eLearning unit “Fremde Federn Finden” (2007). This may skew the results in favor of those systems that only check a small portion of a text.

For all of the test cases, permission from the copyright owner was obtained to use their texts in this manner. For some texts that would have been interesting, it was not possible to obtain permission. It was important to have new test cases for this test, as a number of the old test cases are either stored in the databases of some of the systems or have been plagiarized by others. As the deadline for starting the test approached, we ended up using proportionately more text from the Wikipedia (eight of the new cases) as had originally been planned. The 20 new test cases were written by Matthias Zarzecki, a student at the HTW Berlin, using different forms of plagiarism that were specified in advance.

In addition to the new test cases, a few of the older ones were used, in particular in order to see if systems that had been tested previously still kept copies of the papers. Test case #33 was also used as it includes a number of diacritics and the special Icelandic character 'ð' (eth). A number of cases that were examined but not included in the numerical evaluation were also used. In addition, a large test case was constructed by generating random text and inserting plagiarized text from the smaller test cases. This was done in order to see if the systems could detect the same plagiarism both in a short and in a long text. One test was 40 pages long, simulating a bachelor's thesis, and one was 80 pages long, simulating a master's thesis.

A test case in Japanese had been included in the 2010 test, this time a text in Hebrew, taken from the Hebrew Wikipedia and constructed by Prof. R. H. Belmaker, M.D. from the Ben Gurion University of the Negev, was used. He also provided us with a scientific paper in Hebrew in order to see if the systems were able to work with longer texts in Hebrew.

Homoglyphs, letters that look the same but have different internal representations, are reported in student forums to be a method for foiling plagiarism detection systems. The students just replace all letters, for example the letter 's' or 'e', with a letter from a non-Latin alphabet that looks the same. There are more than 40 substitutions possible. Thus, some test cases were also constructed that used homoglyphs in order to see if the systems were able to detect this usage. These tests were also not part of the numerical evaluation. Only Turnitin was able to find the source despite use of the homoglyph substitution, and Urkund at least registered the use of non-Latin letters, even if it was unable to find the Wikipedia source.

Additional test cases were constructed that used a source findable through Google Books or that used a scanned text for which OCR-recognition had already been done on the PDF. The system Turnitin was given a specific test – since they offer CrossCheck to publishers and announce that they can find plagiarism from sources that are stored in this scientific journal database, one test case was constructed that plagiarized a scientific paper found in that database.

The types of the test cases and the specifications for each is given in Section 2.

1.3 Evaluation criteria

Before the test began, a rubric was created for scoring how effective the systems were at finding plagiarism. In the past, the HTW tests have used either a binary (0/1) or a four-level scale (0-3) for grading. Since in the past half points were often given, it was decided to use a six-level grading system (0-5).

An example rubric for a copy & paste plagiarism such as #42 that is 92% plagiarized (664 out of 715 words are copied):

5 points = > 75% plagiarism detected, relevant source named
4 points = 50-75% plagiarism detected, relevant source named
3 points = 25-49% plagiarism detected, relevant source named
2 points = < 25% plagiarism detected, relevant source named
1 point = plagiarism detected, irrelevant source named
0 points = no plagiarism detected

If the test case consisted of multiple sources, the rubric may have looked like this:

5 points = both sources given with > 40% plagiarism
4 points = both sources given with > 20% plagiarism
3 points = only one source given with > 40% plagiarism
2 points = only one source given with > 20% plagiarism
1 point = only one source given with < 20% plagiarism
0 points = no plagiarism detected

For original texts, 5 points were awarded for reporting no plagiarism. If only less than 5% plagiarism was reported, 3 points were given. Anything higher than that would register as a false positive resulted in no points being given.

For test cases #66 and #67, if the systems found plagiarism when only a small text was submitted but did not find the exact same plagiarism when it was surrounded by much other text, points were deducted, up to 5 points in total.

As it turned out, this sort of rubric muddled what one would expect of an effective system. The percentages returned by many the systems are first and foremost completely arbitrary numbers. Very seldom is it possible to determine why a system is reporting the percentages it does. The rubric detailed above assumed that the values reported were reliable numbers.

We were sometimes perplexed by links being returned that were registering a high amount of overlap. But as some systems only return a link and don't mark the text parallels, we were not able to see and confirm the overlap. Other systems would report plagiarism from pages that were no longer available on the Internet. Thus, if a system registered many possible sources with high values, they would be awarded more points. A system that only returned one source, the correct one, but with a smaller amount of plagiarism would end up with fewer points.

This cannot be regarded as accurately representing what the users of the system would probably consider to be good practice. As a result, the numbers returned by the system were still used to rank the systems in a general manner, as given in section 3.1, but they cannot be considered to be absolute indicators of the effectiveness of the systems. In particular, it is not possible to declare one system better or worse with respect to the others in the group, and not even the top-scoring systems can be recommended for general use. In particular, it must be seen that the maximum number of points for the evaluation was 130. The system that obtained the most points only reached 73% of this figure, which is not a good grade. Five systems did not even manage to get 50% of the possible points. Summing the maximum points earned by any system for each test case would result only in 102 points, since plagiarism by translation is not found, as well as copies from Google Books in general. Still, we kept the maximum at 130, as it is important that all kind of plagiarism be detected and not just the kinds easy to spot. The grouping will be discussed in detail in section 3.

1.4 Procedures

The test cases were prepared in a variety of formats: PDF, DOC, and TXT. There was also a ZIP archive prepared of each of these formats, as we wanted to see if the systems could deal with uploading multiple files at once, or if each file had to be uploaded individually. The first attempt for each system was always to have them use the PDF files. If that did not work, then DOC or TXT was tried. For each format the first attempt was to get the system to accept a ZIP archive, as an educator with multiple files to check would prefer to upload just one file and not each individually. If it was impossible to upload a ZIP archive, the files were submitted one by one. If the system only offered a field into which text had to be copied, then the text from the TXT-version was used. We attempted to copy the entire text into the window at once, which was not always possible. Some systems cut off the text without warning.

The general principle for the test is to use the default setup of the system. Many offer various parameters that can be set in order to fine-tune a search. Since it is not always clear what exactly each of the parameters do, the default is the one that is assumed to be used by a majority of educators. The only exception to this rule was an attempt that was made on our part to locate and enable the property of **not** storing the paper in the system's database, as we often do not have the permission of the student to give a copy of the paper to a third party for their arbitrary and irrevocable use, as dictated by the terms of use in many systems. Most systems that store text will do so automatically in the default setup. It was not always easy to deduce how exactly to go about avoiding this. This is one of the major criticisms of plagiarism detection software: That they do not make clear exactly what is being done with the texts and who can access them.

With the help of a pseudonymous email account, we asked an honest support question, addressing it to the official support address. If a telephone number was given on the web site, we called during German office hours, as the assumption is that for professional use, an answer within 24 hours by email or a phone being answered immediately during normal working hours is expected. Many systems do not even offer a telephone number or an email address, but just have an online form that can be filled out.

After the effectiveness test, a usability checklist with 27 desired properties of a plagiarism detection system was filled out for each system. These include properties such as being able to store reports, having a side-by-side view, consistent use of German, prompt answer to support question, etc. At the end the testers discussed among themselves and then agreed on an overall subjective usability grade on a scale of 0-15, with 15 being the best.

2. Test cases

This section will give details about the composition of the test cases that were used in the 2013 test. Since one of major aspects is to test various types of plagiarism, first a typology of plagiarism will be briefly presented. Then the overall makeup of the new test cases 2013 will be discussed, and then the test cases will be listed.

2.1 Types of plagiarism

Weber-Wulff & Wohnsdorf (2006) defined a typology of plagiarism that is the basis for constructing the test cases for testing plagiarism detection systems. During the work on plagiarism in doctoral dissertations (GuttenPlag Wiki 2011; VroniPlag Wiki 2013) various special forms have been discovered and described, but the tests will be focussing on these types.

- **Copy & Paste**
  This is more or less the only kind of plagiarism that is quickly recognizable and universally agreed on to be plagiarism. The plagiarist locates a useful source and copies a portion of that, perhaps with a few minor changes, into the text that is to be

submitted as own work. Occasionally, an entire work is copied in this manner, only changing the name of the author.

- **Disguised Plagiarism**
  We speak of disguised plagiarism when text from a source is copied and then some effort is exerted in order to disguise the copy. Words may be deleted or inserted, word orders or verb forms changed, or even an attempt at paraphrase may be undertaken. However, since no source is given, or only given for a part of the text taken, this is still considered to be plagiarism.
- **Plagiarism by Translation**
  When a text is taken from one language and translated, either manually or with the help of an automatic translation system, and used without the source being named, then we speak of plagiarism by translation.
- **Shake & Paste**
  Among students a variation of copy & paste can often be seen whereby paragraphs are taken from a number of different sources and compiled, often without a sensible order. Each paragraph will be well written in and of itself, but there is no clear transition from one paragraph to the next. When this is done on the level of snippets, that is parts of sentences "glued" together, we sometimes speak of *mosaic plagiarism*.
- **Structural Plagiarism**
  Taking the idea of someone else, their chain of arguments, their selection of quotations from other people, or even the footnotes that they use in the same order without giving credit is considered to be structural plagiarism. This type of plagiarism is quite difficult to determine, as one must read both texts very closely to see what has been taken.
- **Pawn Sacrifice**
  Benjamin Lahusen (2006) described a sort of plagiarism he had found in which the plagiarist does give a reference or even a proper quotation, but does not note that the text continues on far beyond the citation, or in which the plagiarist uses the exact wording of the source without any indication that this is, indeed, a word-for-word quotation.

2.2 Test Case Variations

A wide variety of test cases were used for the 2013 test. There were 35 test cases used, although only 26 were included in the evaluation, with two additional cases potentially deducting up to 5 points. The numbering of the cases continued from the previous ones, as occasionally the old ones are reused in a test. The makeup of the test cases is as follows:

- Four test cases from previous tests were used (#21, #33, #34, #36)
- 20 new test cases (#42-#61) were constructed as follows
  - Five of the new test cases are in English, the rest are in German; **the English cases are marked in boldface.**
  - Four original texts (#50, #56, #59, **#60**)
  - One translation (#44)
  - One Google Books plagiarism with an entire page as the source (#55)
  - One Google Books plagiarism using only snippets (#61)
  - One pawn sacrifice (**#45**)
  - Two copy & paste plagiarisms (#49, #57)

    - Six disguised plagiarisms (**#42**, **#45**, #48, #52, #54, #58) and one case disguised using a synonymizer **(#46**) that substituted 25 % of the text with synonyms from an online thesaurus
    - Four shake & paste plagiarisms (#43, #47, **#51**, #53)
    - Wikipedia was used as a source for eight of the new cases (#43-de, #45-en, #46-en, #51-en, #52-de, #57-de, #58-de, #63-he) and was used for three of the older ones reused (#21-de, #33-en, #36-fr)
- Two additional test cases were in Hebrew, a copy & paste plagiarism from the Hebrew Wikipedia (#63) and a medical journal article (#62), these were not given numerical point scores, they were just used to see how the systems react to right-to-left writing systems and non-Latin characters, and to see if the Wikipedia source can be found.
- One medical journal article in English (#64) was included that was also not evaluated, it was only included as filler material.
- One plagiarized doctoral dissertation (#65) in German from 1912 consisting of 91 pages of PDF. Optical character recognition was performed on the thesis in order to make it easier for the systems to work with, three sources for this thesis are easily found using Google Books. There was one point given for finding each of the sources, and since one source was responsible for most of the thesis, more points were planned for finding more of the source.
- Two additional test cases (#66, #67) were prepared by generating 40 and 80 pages of random sentences and then injecting text from the new test cases in order to simulate a bachelor's and a master's thesis. #66 was injected with larger paragraphs, #67 with sentences from plagiarized texts. If the plagiarisms could not be found in the larger text but were found in the smaller one, points were taken off.
- One additional test case (#68) was a ZIP file with 5 student papers from 2001 stored as PDFs with known plagiarism. Additional points were given for finding any of this plagiarism.
- One additional test case (#69) was prepared as a disguised plagiarism of a paper in a scientific journal on library science. It was only used to test Turnitin's claim that it is able to identify such plagiarisms (it can).
- Three additional test cases (#70, #71, #72) were constructed using homoglyphs in order to see if systems were able to still recognize plagiarism if it was disguised by replacing some characters with ones that look the same but are actually encoded differently. These cases were also not part of the numeric evaluation.

2.3 List of test cases

The following table lists the numbers and names of the test cases and gives a short description of the test case. If the test case was included in the numerical evaluation, the column **E** will contain a '+' for points accrued or a '–' for points deducted. The language is given in column **L** (DE = German, EN = English, HE = Hebrew), as well as a list of the sources used. In the online version of this document, the sources and test cases will be linked.

Table 1: Test Cases 2013

Test Case	Description	E	L	Sources
21-Tibet	A shake & paste plagiarism from the 2010 test. It uses three sources.	+	DE	1. Süddeutsche Zeitung 2. Computerwoche 3. Wikipedia Tibetische Unruhen 2008
33-Eyjafjallajoekull	A copy & paste plagiarism from the Wikipedia, the article has many Icelandic characters in it.	+	EN	1. Wikipedia Ejafjällajökull
34-Stieg-Larsson	Original text from the 2010 test.	+	DE	Original
36-Champagne	Plagiarism by translation of excerpts of the French Wikipedia article about champagne bottle sizes that should be findable because of the series listing the size names. Google Translate was used, with some polishing.	+	EN	1. Wikipedia Champagne (AOC)
42-Arduino	Disguised plagiarism from one source.	+	EN	1. Tronixstuff
43-Brüder-Grimm	Shake & paste plagiarism with two sources.	+	DE	1. Wikipedia Brüder Grimm 2. Wikisource
44-Holy-Grail	Plagiarism by translation from the source named, using Google Translate. Many portions were skipped so that it is not 1:1.	+	DE	1. Newadvent
45-Strelitzia	A disguised plagiarism with a pawn sacrifice. The text was highly disguised and footnotes were added.	+	EN	1. Wikipedia Strelitzia
46-Thermos-kanne	A text from the English Wikipedia was changed using the automatic Plagiarisma Synonymizer set to change 25% of the text.	+	EN	1. Wikipedia Vacuum flask
47-Tessellation	Shake & paste plagiarism of four sources. Some portions were skipped, a series was re-ordered, and different fonts were used.	+	DE	1. Mathematische Basteleien 2. Hartware 3. Schoenleber 4. Vismath
48-Berliner-Baer	A disguised plagiarism with some original material.	+	DE	1. Zeit für Taten
49-Betamax	Copy & paste plagiarism with some original material at the beginning and end, some source text skipped.	+	DE	1. Betamax
50-Union-Jack	An original text that references the Wikipedia properly.	+	DE	Original

51-London-Blitz	Shake & paste plagiarism, the paragraphs are well shaken so that the text makes little sense.	+	EN	1. Wikipedia The Blitz 2. Eyewittnesshistory 3. Guardian 4. 20centurylondon
52-Boxer-Rebellion	A disguised plagiarism from the German Wikipedia.	+	DE	1. Wikipedia Boxeraufstand
53-Falkland-Krieg	A shake & paste plagiarism from two sources.	+	DE	1. Die Presse 2. Wissen.de
54-Südpol	An intensively disguised plagiarism.	+	DE	1. Helles Köpfchen
55-Paul-Englisch	The oft-repeated plagiarism definition by Paul Englisch was combined with material from a book that is either available at Google Books, or without OCR at the Visuallibrary.	+	DE	1. Google Books 2. Visuallibrary
56-Hoover-Dam	An original text.	+	DE	Original
57-Fallingwater	A copy & paste plagiarism from the German Wikipedia.	+	DE	1. Wikipedia Fallingwater
58-Phillip-K-Dick	An intensively disguised plagiarism from the German Wikipedia.	+	DE	1. Wikipedia Phillip K. Dick
59-Alpha-Centauri	An original text.	+	DE	Original
60-Rolltreppe	An original text with proper references to the sources.	+	EN	1. Howstuffwork 2. Wikipedia Escalators 3. inventors
61-Wasser-wirtschaft	A short excerpt from the dissertation “Entwickelungsfragen der Wasserwirtschaft” (1912), findable with Google Books.	+	DE	
62-Hebrew-Med	A medical journal article in Hebrew.		HE	
63-Hebrew-Plag	A copy & paste plagiarism from the Hebrew Wikipedia entry on plagiarism.		HE	1. Wikipedia גניבה ספרותית
64-Medical	A medical journal article in English.		EN	
65-Dissertation	This is the complete doctoral dissertation “Entwickelungs-fragen der Wasserwirtschaft” (1912), 91 pages of PDF with OCR. At least 3 sources are findable with Google Books.	+	DE	
66-Random-Paragraphs	40 pages of random sentences generated from a dictionary with paragraphs from the test cases 21-62 injected.	-	EN, DE	

67-Heavy-Random	80 pages of random sentences generated from a dictionary with sentences from the test cases 21-62 injected.	-	EN, DE	
68-Papers	Five student papers in PDF without OCR, four are known and one suspected plagiarisms from 2001.	*	EN, DE	
69-Library-Hi-Tech	Excerpt from a journal article that is normally behind a paywall, disguised.		EN	from Library Hi Tech Vol. 31 No. 1, 2013 pp. 5–7, DOI 10.1108/07378831311310338
70-Hosen-vergleich	Three paragraphs from two entries from the German Wikipedia with some characters replaced by Cyrillic and Greek homoglyphs.		DE	1. Wikipedia 1 Material & Bekleidung 2. Wikipedia 2 Reithose
71-AOC02	41 Cyrillic and Greek homoglyphs were used on an entry from the German Wikipedia.		DE	1. Wikipedia Appellation d’Origine Contrôlée
72-Kafka	Two texts from Kafka that are in the public domain were first collected to a shake & paste plagiarism in an A - B - A – B format for 60 pages, then Cyrillic and Greek homoglyphs were used to replace 41 characters.		DE	A: “Das Schloß” B: “Der Prozeß”

3. Results

This section is devoted to presenting the results of the 2013 plagiarism detection system test and discussing the problems encountered during the test. Recommendations will be given in the next section, and the individual results are available in an appendix and online at the Plagiarism Portal.

3.1 Evaluation results

In previous tests, the results of the numerical evaluation were summed up, a ranking table was created, and the systems were grouped into useful, partially useful, marginally useful, and useless systems for adoption in an educational setting. There are – in previous tests and now – no systems that can be recommended as useful, as there are too many instances in which the systems fail.

As discussed in section 1.3, the rubric selected for evaluating the systems in 2013 rewarded those systems that reported high values of plagiarism, even though for reasons such as outdated links they were not able to substantiate that value. Systems that returned massive amounts of irrelevant links which served only to inflate the impression of plagiarism had to be given more points than more conservative systems that did report plagiarism and gave the correct source. The reason for that was that it was practically impossible to evaluate the relevance of all of the links. Some systems even returned a different answer when asked to re-evaluate a paper at a later point in time. From the point of view of an educational institution, this is counter-intuitive. And as we have realized, it is not important in an educational setting to find all of the plagiarism in a paper. It is sufficient to find enough for a sanction to be necessary.

The following table lists the point values awarded, although it should not be considered an absolute ranking for which a system can advertise "best in test". Rather, it shows a relative ranking for effectiveness that must be considered together with the usability aspects. There are two columns given for this, one is for the number of properties on the usability checklist that were visible in the product, and the second column is a subjective usability score that represents the subjective feeling the testers had for how well the system works in an academic context.

Table 2: Numerical Results

Number	Test System	Effective-ness	Percent	Usability Checklist	Subjective Usability Score
S13-06	Urkund	95	73%	12.5	10 (C+)
S13-03	Turnitin	87	67%	15.5	12 (B)
S13-19	Copyscape	87	67%	15	7 (D+)
S13-05	Ephorus	76	58%	19	9 (C)
S13-01	PlagAware	75	58%	19	11 (B-)
S13-18	Strike Plagiarism	75	58%	17	10 (C+)
S13-07	PlagScan	72	55%	17	9 (C)
S13-08	Compilatio	72	55%	15	4 (F)
S13-13	PlagiarismDetect Premium	72	55%	12	5 (D-)
S13-04	Docoloc	70	54%	13	4 (F)
S13-13	PlagiarismDetect Standard	65	50%	12	5 (D-)
S13-12	Duplichecker	63	48%	12	5 (D-)
S13-17	PlagTracker	41	32%	12	7 (D+)
S13-02	Plagiarisma	39	30%	7	2 (F)
S13-09	OAPS	39	30%	11	6 (D)
S13-10	PlagiarismFinder	38	29%	19	11 (B-)
		Max: 130		Scale: 1-27	Scale: 1-15 (Letter grades)

Legend (% of total points, according to the ECTS grading scale):

Very Good	90% or higher
Good	80-89%
Adequate	70-79%
Poor	60-69%
Unacceptable	Under 60%

3.2 Discussion of Results

The grading of the systems was done according to the ECTS grading categories that are used in universities for assigning grades to students. A “very good” is given to systems with 90% or more of the possible points, anything below 60% is considered unacceptable.

There are three systems in the “partially useful” category, Urkund, Turnitin, and Copyscape. While Urkund received a few more points than the other two systems, there were still some usability issues and the amount of points would still only be considered “adequate” on the ECTS scale. Turnitin was given a “good” overall usability grade, while Copyscape only scored “poor” on this aspect. All three systems, however, did not fare very well on the usability checklist.

The second group, the marginally useful systems with only between 48% and 58% effectiveness, includes eight systems. PlagAware scored “good” on the subjective usability and Ephorus, PlagScan, and StrikePlagiarism were deemed “adequate” in that respect. With regards to the usability checklist, Ephorus and PlagAware reached the “adequate grade”, with StrikePlagiarism and PlagScan passing with a “poor” mark.

The last group, the systems deemed useless for academic purposes, found practically no plagiarism, even if the systems such as PlagiarismFinder were actually graded “good” with respect to the usability.

Because of this extremely mixed result, it is not possible to recommend the use of a particular system, most particularly as there are many different use cases for the various systems and some are particularly useful for specific purposes, but not generally.

3.3 Problems encountered

Every system suffers from two major problems that have to do with the usage context of the systems. For academic work, both can have catastrophic results.

- **False positives** happen when systems report significant plagiarism scores for original work. This can happen if the text uses many common phrases and the system reacts to four or five words in sequence as being plagiarism without examining a wider context. For example Compilatio reported that case #34 (an original text) was 11% plagiarized. At least it listed the phrases that it found troublesome: “Stieg Larsson was born in 1954”, “The rest of his childhood he lived” and “For the next birthday he got a”. An educator can quickly see that this is not plagaiarism. PlagTracker reported that case #59 was 77% plagiarized, giving a copy of a Douglas Adams book that does mention Alpha Centauri and the Wikipedia entry on Mars (!) as the sources. A careless teacher might see the large number and the Wikipedia listed as a source and too quickly draw a false conclusion, although there are only tiny phrases such as “Es wird vermutet, dass sich” that are the same in both texts.

  One reason that false positives are such a problem in academic contexts is that some schools have carelessly set an official threshold, above which a sanctioning process is set in motion. As we have seen in our tests, each software system returns different results, and very rarely hits the amount of plagiarism exactly because such a number is difficult to calculate when words are added or removed. Another problem with false positives is the damage done to a promising student who is facing an incorrect accusation of plagiarism. It places quite a strain on the teacher-student relationship.

- **False negatives** happen when the systems do not find plagiarism that is in the texts. There are a variety of reasons for this. Many systems only check a sample of the text to be investigated; some have problems with umlauts or are confused by homoglyphs; some only check for exact copies and miss disguised plagiarism; most systems cannot deal with plagiarism from books or scientific journal articles that are not available digitally. The problem with false negatives is that a student gets away with a plagiarism. They may brag to fellow students and explain how they disguised their text. Or, as has been the case with numerous doctoral dissertations in Germany, the plagiarism is not discovered until later, when the book is available in print or as an eBook.

These two, systemic problems with so-called plagiarism detection software are the reasons why it can only be considered to be a tool, not some sort of automatic determination of plagiarism.

Additional problems that were encountered in the test include:

- Online publication of the test results with simple numeric URLs made it easy to guess valid test numbers and see the results of other students at other schools – after the company was informed, they removed this feature immediately.
- Quite a number of systems report plagiarism with irrelevant or no longer valid links.
- Many systems do not report the Wikipedia as a source, but one of the many copies of the Wikipedia. This can be problematic if the site reported is using the Wikipedia text (and that is legal if the provisions of the CC-BY-SA license are followed) to promote other services such as erotic media sales that might not be compatible with use on university computers.
- The language use is not always consistent in many systems, they will either revert to their original language at times, or use translated terms that do not make sense.
- Some systems only permit one text to be examined at a time. For use as a tool to check one suspicious paper, that is okay. But when a number of papers need to be examined, the educator is forced to spend much time watching for when the system is finished with one so that the next one can be checked.
- Many clicks are sometimes necessary for checking just one file. For example, StrikePlagiarism insists on accepting the terms of use for each file that is uploaded. On other systems the tab for the plagiarism test must be re-visited for every test, one needs a number of clicks to select the file and start the test, and then many mores clicks are necessary to find the report.
- The reports, if any are made, are often difficult to interpret.
- A price comparison is impossible to make because the systems have such different functionality and not all companies publish their prices. Instead, they will meet with a representative of the university and agree on a price, usually based on a price per student per year.

4. Recommendations

This final section contains some recommendations both for universities contemplating using plagiarism detection software and for the companies that market such software.

4.1 Universities

Universities should focus their efforts more on plagiarism avoidance than on after-the-fact detection and punishment. But for those cases in which an educator has a suspicion of plagiarism, the university should provide some means of using such systems as a tool. The teachers should be educated in the use of search machines in order to discover plagiarism on their own. But for more disguised cases, there should be a university service that offers help in finding possible sources. Such a service would be best if it were offered by the university library or the university computer center. Both organizations are independent of the departments and schools and have personnel skilled in the use of complex computer programs. The library would actually be the best place to start a plagiarism awareness center, as they can be offering courses on avoiding plagiarism and doing proper research, as well as helping with plagiarism detection. However, this cannot be just an added duty for the librarians currently employed, but there must be sufficient resources available to provide the assistance needed.

If a university decides to purchase software, it would do well to purchase access to at least two systems, as the systems will give different results. Using multiple systems increases chance of finding a source for a suspected plagiarism. Using a system to screen all papers is problematic, however, because of the false positives and false negatives that occur, as discussed in section 3.2. It might be useful as a general screening of all first-year papers, and the results can be used in a formative manner to teach students good scientific writing. But beyond that, the problems outweigh the perceived benefits, because there is still no system that can replace the manual finding of plagiarism in texts.

It is probably a good idea for universities to investigate a sample of their past published doctoral theses, perhaps as a training ground for the personnel in a plagiarism awareness center. That would give experience in using the systems with large files, and perhaps filter out the odd plagiarism that might cause embarrassment in the future.

4.2 Software companies

The software companies need to understand that educators want a tool to make their life easier. In particular, this relates to the preparation of reports. An examination board that will be investigating a matter of plagiarism will want to see a synopsis, a side-by-side documentation with the plagiarism on one side and the source on the other side of the paper, preferably lined up (or marked in color) so that it is trivial to see where the plagiarism is. It should be made as easy as possible for an educator to mark up the report or at least add comments.

Make it crystal clear if a paper is being stored or not or not! If papers are being stored and there are different options (only for one school, only for one area, generally available) it should be made clear at all times, perhaps with an icon, if this paper is stored. It is clear that the companies want to keep copies, but they should be fair and explicit about what is being done.

Watch out for the “tricks” students pass around for avoiding plagiarism. They include using homoglyphs, substituting a character in white for the spaces, automatically replacing words with synonyms, etc. It is also important to be aware of diacritical marks that are used in the various languages. On the other hand, marking simple, often used phrases as plagiarism will contribute to false positives, which are highly undesirable in university work.

It would be useful for the universities to be able to purchase bundles of plagiarism tests for a set fee. Concepts such as “PlagPoints” or “Credits” that are used for a unit such as 500 words are very good for a university that wants to get started with using such a system, and the costs are much better to plan for. A university may only have a certain amount of money available and not be able to commit to a subscription model. Companies are passing up an opportunity to earn money if they are not offering what the customers would like to purchase. There should always be a possibility for using a test system, either as a free trial or a reduced fee, so that the universities can see if such a system is usable for their purposes.

It is vital for companies to offer professional service. This will include offering a telephone service, rapid email support, and not offering ghostwriting or “editing” as a side-offer.

4.3 Summary

So-called plagiarism detection software does not detect plagiarism. In general, it can only demonstrate text parallels. The decision as to whether a text is plagiarism or not must solely rest with the educator using the software: It is only a tool, not an absolute test.

In particular, users must be aware of the false positive and false negative problems that all systems have. A university can and should make software available for their educators to use, but they should not use it as a general screening tool for all texts. If at all, general screening could only be reasonably used for first-year student papers.

5. Bibliography

GuttenPlag Wiki (2011) http://de.guttenplag.wikia.de/wiki

Lahusen, B. (2006) Goldene Zeiten: Anmerkungen zu Hans-Peter Schwintowski, Juristische Methodenlehre, UTB basics Recht und Wirtschaft 2005. In: *Kritische Justiz* , Vol. 39, No. 4, pp. 398–417. Available at http://www.kj.nomos.de/fileadmin/kj/doc/2006/20064Lahusen_S_398.pdf

Portal Plagiarism (n.d.) http://plagiat.htw-berlin.de (in German, with a section in English on the plagiarism detection software tests)

VroniPlag Wiki (2013) http://de.vroniplag.wikia.de/wiki

Weber-Wulff, D. (2008) On the utility of plagiarism detection software. Third International Conference on Plagiarism, Newcastle on Tyne.

Weber-Wulff, D. (2009) Fremde Federn Finden - Plagiatserkennungssysteme im Vergleich. In: *Die Neue Hochschule.* No. 2-3, pp. 40–47.

Weber-Wulff, D. (2014) *False Feathers: A Perspective on Academic Plagiarism*. Heidelberg: Springer Verlag. (In press)

Weber-Wulff, D. & Köhler, K. (2011a) Kopienjäger - Cloud-Software vs. menschliche Crowd in der Plagiaterkennung. In: *iX*, No. 6, pp. 78–82.

Weber-Wulff, D. & Köhler, K. (2011b) *Plagiatserkennungssoftware 2010*. In: *Information : Wissenschaft und Praxis* Vol. 62, No. 4, pp. 159–166.

Weber-Wulff, D. & Pomerenke, M. (2007) Plagiat 2.0 - Was taugen die Anti-Abschreiber Programme? In: *Spiegel-Online*, 27 September.

Weber-Wulff, D. & Wohnsdorf, G. (2006) Strategien der Plagiatsbekämpfung. In: *Information: Wissenschaft & Praxis.* Vol. 57, No. 2, pp. 90–98.