Detekce plagiátů

Detekce plagiátů je proces vyhledání případů plagiátorství v rámci práce nebo dokumentu. Rozsáhlé používání počítačů a příchod internetu usnadnily plagiátorství práce ostatních. Většina případů plagiátorství se nachází v akademické sféře, kde jsou obvykle dokumenty nebo zprávy. Plagiátorství však lze nalézt prakticky ve všech oblastech, včetně románů, vědeckých prací, výtvarných návrhů a zdrojového kódu.

Detekce plagiátorství může být buď manuální nebo softwarově podporovaná. Ruční zjišťování vyžaduje značné úsilí a vynikající paměť a je nepraktické v případech, kdy je třeba porovnat příliš mnoho dokumentů nebo nejsou k dispozici srovnávací dokumenty. Detekce pomocí softwaru umožňuje vzájemné porovnání rozsáhlých sbírek dokumentů, čímž je úspěšnější detekce mnohem pravděpodobnější.

Praxe plagiátorství použitím dostatečných slovních substitucí k vyloučení detekčního softwaru je známá jako rogeting.[1]

Detekce pomocí softwaru

Počítačová detekce plagiátů (CaPD) je úloha získávání informací (IR) podporovaná specializovanými IR systémy, označovanými jako systémy detekce plagiátů (PDS).

V textových dokumentech

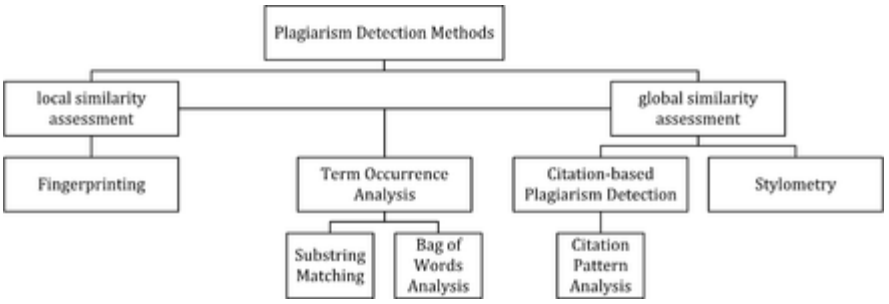
Systémy detekce textu a plagiátorství implementují jeden ze dvou obecných detekčních přístupů, z nichž jeden je vnější, druhý je vnitřní [2]. Externí detekční systémy porovnávají podezřelý dokument s referenční sbírkou, což je soubor dokumentů, o nichž se předpokládá, že jsou pravé. [3] Na základě vybraného modelu dokumentu a předem definovaných kritérií podobnosti je úkolem detekce získání všech dokumentů, které obsahují text, který je podobný stupni nad vybranou prahovou hodnotou, do textu v podezřelém dokumentu. [4] Vlastní PDS analyzuje pouze text, který má být vyhodnocen, aniž by došlo k porovnávání s externími dokumenty. Cílem tohoto přístupu je rozpoznat změny v jedinečném psacím stylu autora jako ukazatele potenciálního plagiátorství. [5] PDS nejsou schopny spolehlivě identifikovat plagiátorství bez lidského úsudku. Podobnosti jsou vypočítávány pomocí předdefinovaných modelů dokumentů a mohou představovat falešné pozitivy. [6] [7] [8] [9] [10]

Efektivita v oblasti vysokoškolského vzdělávání

Byla provedena studie o testování účinnosti softwaru pro detekci plagiátů v prostředí vysokoškolského vzdělávání. Jedna část studie přidělila skupině studentů, aby napsali příspěvek. Tito studenti byli nejprve informováni o plagiátovi a informováni o tom, že jejich práce by měla probíhat prostřednictvím systému detekce plagiátů. Druhá skupina studentů byla přiřazena k psaní příspěvku bez informací o plagiátovi. Vědci očekávali, že v první skupině najdou nižší sazby, ale v obou skupinách zjistili zhruba stejnou míru plagiátorství.[11]

Přístupy

Níže uvedený obrázek představuje klasifikaci všech přístupů detekce, které se v současné době používají pro detekci plagiátorství pomocí počítačů. Přístupy jsou charakterizovány typem hodnocení podobnosti, které provádějí: globální nebo místní. Globální přístupy pro posuzování podobnosti využívají charakteristiky převzaté z větších částí textu nebo z dokumentu jako celku, aby vypočítali podobnost, zatímco místní metody zkoumají jako vstupy předem vybrané segmenty textu.



Klasifikace metod detekce plagiátů s počítačem

Fingerprinting

Fingerprinting je v současné době nejpoužívanějším přístupem k detekci plagiátů. Tato metoda vytváří reprezentativní digesty dokumentů výběrem souboru více podřetězců (n-gramů) z nich. Sady představují otisky prstů a jejich prvky se nazývají markery. [12] [13] Podezřelý dokument je zkontrolován pro plagiátorství pomocí výpočtu jeho otisků prstů a dotazování detailů s předkompilovaným indexem otisků prstů pro všechny dokumenty referenční sbírky. Minutia, které se shodují s ostatními dokumenty, označují segmenty sdílených textů a naznačují možný plagiát, pokud překročí zvolenou hranici podobnosti. [14] Výpočtové zdroje a čas jsou omezujícími faktory pro otisky prstů, což je důvod, proč tato metoda typicky porovnává pouze podmnožinu detailů, aby urychlila výpočet a umožnila kontroly ve velmi rozsáhlých sbírkách, jako je například internet.[12]

Shoda řetězce

String matching je běžný přístup používaný v informatice. Při použití problému detekce plagiátů se porovnávají dokumenty pro doslovné překrývání textu. Pro řešení tohoto úkolu byla navržena řada metod, z nichž některé byly přizpůsobeny externí detekci plagiátorství. Kontrola podezřelého dokumentu v tomto nastavení vyžaduje výpočet a uložení efektivně srovnatelných reprezentací pro všechny dokumenty v referenční sbírce, aby byly porovnány po páru. Obecně platí, že pro tento úkol byly použity příponové modely dokumentů, například příponové stromy nebo přípony. Nicméně, podřetězec shody zůstává výpočetně drahý, což z něj činí neživotaschopné řešení pro kontrolu velkých sbírek dokumentů.[15][16][17]

Sáček slov

Analýza pytlů slov představuje přijetí vyhledávání vektorových prostorů, tradičního IR konceptu, do oblasti detekce plagiátorství. Dokumenty jsou reprezentovány jako jeden nebo více vektorů, např. pro různé části dokumentu, které se používají pro výpočty podobnosti v páru. Výpočet podobnosti se pak může opírat o tradiční měřítko podobnosti kosinů nebo o sofistikovanější podobná opatření.[18][19][20]

Citační analýza

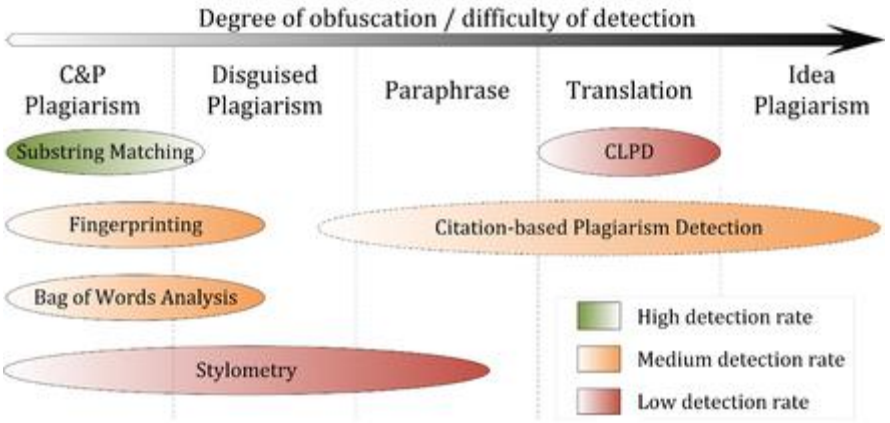
Citace založená plagiátorová detekce (CbPD) [21] se opírá o citační analýzu a je jediným přístupem k detekci plagiátů, který se nespoléhá na podobnost textu. [22] CbPD zkoumá citace a referenční informace v textech k identifikaci podobných vzorů v citačních sekvencích. Jako takový je tento přístup vhodný pro vědecké texty nebo jiné akademické dokumenty, které obsahují citace. Citační analýza k odhalení plagiátorství je poměrně mladá koncepce. To nebylo přijato komerčním softwarem, ale existuje první prototyp systému citací založeného na plagiátorství. [23] Podobná pořadí a blízkost citací ve zkoumaných dokumentech jsou hlavními kritérii používanými pro výpočet podobností citačních vzorů. Citační modely představují subsekvence, které neobsahují výlučně citace sdílené porovnávanými dokumenty. [22] [24] Faktory zahrnující absolutní číslo nebo relativní zlomek sdílených citací ve vzorku, stejně jako pravděpodobnost, že citace se vyskytují společně v dokumentu, se také považují za kvantifikaci stupně podobnosti vzoru.[22][24][25][26]

Stylometrie

Stylometrie zahrnuje statistické metody pro kvantifikaci jedinečného psaného stylu autora [27] [28] a používá se hlavně pro přiřazení autora nebo vlastní CaPD. Vytvářením a porovnáváním stylometrických modelů pro různé textové segmenty lze zjistit průchody, které jsou stylisticky odlišné od ostatních, tedy potenciálně plagiátorové.[5]

Výkon

Srovnávací hodnocení systémů detekce plagiátů [3] [29] [30] [31] [32] [33] ukazují, že jejich výkon závisí na typu plagiátorství (viz obrázek). Kromě analýzy citačních vzorů se všechny detekční přístupy opírají o podobnost textu. Je tedy příznačné, že přesnost detekce snižuje počet případů plagiátů, které jsou zmateny.



Detekce výkonu přístupů CaPD v závislosti na typu plagiátorství

Doslovné kopie, např. plagiátorství typu kopírování a vkládání (c & v), nebo spoře skryté případy plagiátorství, lze zjistit s vysokou přesností stávajícími externími PDS, pokud je zdroj softwaru přístupný. Zejména postupy pro porovnávání podřetězců dosahují dobrého výkonu pro plagiátorství c & v, protože běžně používají bezztrátové modely dokumentů, jako jsou příponové stromy. Výkon systémů používajících fingerprinting nebo analýzu balíku slov při detekci kopií závisí na ztrátách informací způsobených použitým modelem dokumentu. Použitím strategií flexibilního zúžení a výběru jsou lépe schopny rozpoznat umírněné formy skrytého plagiátorství ve srovnání s postupy porovnávání podřetězců.

Detekce vnitřního plagiátorství pomocí stylometrie může do jisté míry překonat hranice textové podobnosti porovnáním jazykové podobnosti. Vzhledem k tomu, že stylistické rozdíly mezi plagiovanými a původními segmenty jsou významné a lze je spolehlivě identifikovat, stylometrie může pomoci při identifikaci maskovaného a parafrázovaného plagiátorství. Stylometrické srovnání pravděpodobně selže v případech, kdy jsou segmenty silně parafrázovány až do okamžiku, kdy se více podobají osobnímu psacímu stylu plagiátora, nebo jestliže text byl sestaven několika autory. Výsledky mezinárodních soutěží o detekci plagiátů, které se konaly v letech 2009, 2010 a 2011, [3] [32] [33], stejně jako pokusy prováděné Steinem [34] naznačují, že stylometrická analýza funguje spolehlivě pouze pro délky dokumentů několik tisíc nebo desítek tisíc slov, což omezuje použitelnost metody na nastavení aplikace CaPD.

Zvyšuje se množství výzkumů zkoumající metody a systémy schopné odhalit překládané plagiáty. V současné době se detekce plagiátorství v rámci více jazyků (CLPD) nepovažuje za vyspělou technologii [35] a příslušné systémy nebyly v praxi schopny dosáhnout uspokojivých výsledků detekce. [31]

Použití detekce plagiátů na základě citační analýzy pomocí analýzy citových vzorů je schopno identifikovat silnější parafráze a překlady s vyšší mírou úspěšnosti ve srovnání s jinými detekčními přístupy, protože je nezávislé na textových charakteristikách. [22] [25] Vzhledem k tomu, že analýza citačních vzorů závisí na dostupnosti dostatečných citačních informací, je to omezeno na akademické texty. Zůstává podřazen přístupu založenému na textu při odhalování kratších plagiovaných pasáží, což jsou typické pro případy kopírování a pastování nebo plavání a posunování plagiátů; druhá se týká míchání mírně změněných fragmentů z různých zdrojů.

Software

Návrh softwaru pro detekci plagiátů pro práci s textovými dokumenty je charakterizován řadou faktorů:

Faktor	Popis a alternativy
Rozsah vyhledávání	Ve veřejném internetu pomocí vyhledávačů / Institucionálních databázích / Místní, systémově specifické databázi.
Čas analýzy	Zpoždění mezi odesláním dokumentu a okamžikem, kdy jsou výsledky k dispozici.
Kapacita dokumentů / Dávkové zpracování	Počet dokumentů, které může systém zpracovat za jednotku času.
Kontrola intensity	Jak často a pro jaké typy fragmentů dokumentů (odstavce, věty, řetězce slov s pevnou délkou) se dotazuje systém externích zdrojů, například vyhledávačů.
Typ porovnávacího algoritmu	Algoritmy, které definují způsob, jakým systém používá k vzájemnému porovnávání dokumentů.
Přesnost a zpětné odvolání	Počet dokumentů správně označených jako plagiát ve srovnání s celkovým počtem označených dokumentů a celkovým počtem dokumentů, které byly skutečně plagiovány. Vysoká přesnost znamená, že bylo jako plagiát označeno malé originálních dokumentů a vysoká

úroveň zpětného odvolání znamená, že malé množství plagiátů zůstalo nezjištěno.

Většina rozsáhlých systémů na detekci plagiátů používá velké vnitřní databáze (kromě jiných zdrojů), které rostou s každým dalším dokumentem předloženým k analýze. Tuto funkci však někteří považují za porušení autorských práv studentů.

Ve zdrojovém kódu

Plagiátorství v zdrojovém kódu počítačových programů je také časté a vyžaduje jiné nástroje než ty, které se používají pro porovnávání textů v dokumentu. Významný výzkum byl věnován plagiátorství zdrojového kódu v akademickém prostředí. [37]

Výrazným aspektem plagiátorství zdrojového kódu je to, že neexistují firmy poskytující takovýto typ plagiátorství za úplatu, které lze nalézt v tradičním plagiátorství. Vzhledem k tomu, že většina programových úkolů očekává, že studenti budou psát programy s velmi specifickými požadavky, je velmi obtížné nalézt existující programy, které jim již zadání splňují. Vzhledem k tomu, že integrace externího kódu je často těžší než psát na začátku, většina plagiátorských studentů se rozhodne opisovat od svých spolužáků.

Podle Roye a Cordyho [38] mohou být algoritmy detekce podobnosti zdrojového kódu klasifikovány jako založené na jedné z následujících možností:

- Řetězce - vyhledá přesné textové shody segmentů, například pět slov. Rychlé, ale může dojít ke zmatení přejmenováním identifikátorů.
- Tokeny – tak jako u řetězců, ale pomocí lexikální analýzy, která nejprve převede program na tokeny. Tím se vyhazují mezery, komentáře a jména identifikátorů, což činí systém robustnější než jednoduché nahrazení textu. Většina systémů pro detekci plagiátorství pracuje na této úrovni pomocí různých algoritmů pro měření podobnosti mezi sekvencemi tokenů.
- Derivační stromy – sestaví a porovná derivační stromy. To umožňuje detekovat podobnosti na vyšší úrovni. Například porovnání stromů může normalizovat podmíněné výrazy a detekovat ekvivalentní konstrukty jako navzájem podobné.
- Grafy programové závislosti (GPZ) - GPZ zachycuje skutečný tok řízení v programu a dovoluje lokalizovat mnohem vyšší úroveň rovnocennosti, s vyššími nároky na složitost a dobu výpočtu.
- Metriky - metriky zachycují "skóre" segmentů kódu podle určitých kritérií; například "počet smyček a podmíněných příkazů" nebo "počet použitých proměnných". Metriky lze snadno vypočítat a lze je rychle porovnávat, ale mohou také vést k falešným pozitivním výsledkům: dva fragmenty se stejnými výsledky na souboru metrik mohou dělat zcela jiné věci.
- Hybridní přístupy - například derivační stromy + příponové stromy mohou kombinovat detekční schopnosti derivačních stromů s rychlostí poskytovanou příponovými stromy, což je typ datové struktury odpovídajících řetězcům.

Předchozí klasifikace byla vyvinuta za účelem refaktoringu kódu a ne pro detekci akademického plagiátorství (důležitým cílem refaktoringu je vyhnout se duplicitnímu kódu, označovanému v literatuře jako kódové klony). Výše uvedené přístupy jsou účinné proti různým úrovním podobnosti; nízká podobnost se týká stejného textu, zatímco podobnost na vysoké úrovni může být způsobena podobnými specifikacemi. V akademickém prostředí, kdy

se očekává, že všichni studenti budou kódovat stejné specifikace, se očekává funkčně zcela ekvivalentní kód (s vysokou úrovní podobnosti) a pouze nízká podobnost se považuje za důkaz o podvádění.

Dále se podívejte

- Kategorie: Detektory plagiátorství
- Srovnání softwaru proti plagiátorství
- Místně citlivé hashování
- Hledání nejbližšího souseda
- Detekce parafrází
- Kolmogorov složitost # Komprese – použitý k odhadu podobnosti mezi sekvencemi tokenů v několika systémech

References

1. *Grove, Jack (7 August 2014). "Sinister buttocks? Roget would blush at the crafty cheek Middlesex lecturer gets to the bottom of meaningless phrases found while marking essays". Times Higher Education. Retrieved 15 July 2015.*
2. *Stein, Benno; Koppel, Moshe; Stamatatos, Efstathios (Dec 2007), "Plagiarism Analysis, Authorship Identification, and Near-Duplicate Detection PAN'07" (PDF), SIGIR Forum, **41** (2), doi:10.1145/1328964.1328976*
3. *Potthast, Martin; Stein, Benno; Eiselt, Andreas; Barrón-Cedeño, Alberto; Rosso, Paolo (2009), "Overview of the 1st International Competition on Plagiarism Detection", PAN09 - 3rd Workshop on Uncovering Plagiarism, Authorship and Social Software Misuse and 1st International Competition on Plagiarism Detection (PDF), CEUR Workshop Proceedings, **502**, pp. 1–9, ISSN 1613-0073, archived from the original (PDF) on 2 April 2012*
4. *Stein, Benno; Meyer zu Eissen, Sven; Potthast, Martin (2007), "Strategies for Retrieving Plagiarized Documents", Proceedings 30th Annual International ACM SIGIR Conference (PDF), ACM, pp. 825–826, doi:10.1145/1277741.1277928, ISBN 978-1-59593-597-7*
5. *Meyer zu Eissen, Sven; Stein, Benno (2006), "Intrinsic Plagiarism Detection", Advances in Information Retrieval 28th European Conference on IR Research, ECIR 2006, London, UK, April 10–12, 2006 Proceedings (PDF), Lecture Notes in Computer Science, **3936**, Springer, pp. 565–569, doi:10.1007/11735106_66*
6. *Bao, Jun-Peng; Malcolm, James A. (2006), "Text similarity in academic conference papers", 2nd International Plagiarism Conference Proceedings (PDF), Northumbria University Press*
7. *Clough, Paul (2000), Plagiarism in natural and programming languages an overview of current tools and technologies (PDF) (Technical Report), Department of Computer Science, University of Sheffield, archived from the original (PDF) on 18 August 2011*
8. *Culwin, Fintan; Lancaster, Thomas (2001), "Plagiarism issues for higher education" (PDF), Vine, **31** (2): 36–41, doi:10.1108/03055720010804005, archived from the original (PDF) on 5 April 2012*
9. *Lancaster, Thomas (2003), Effective and Efficient Plagiarism Detection (PDF) (PhD Thesis), School of Computing, Information Systems and Mathematics South Bank University*

10. *Maurer, Hermann; Zaka, Bilal (2007), "Plagiarism - A Problem And How To Fight It", Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2007, AACE, pp. 4451–4458*
11. *Youmans, Robert J. (November 2011). "Does the adoption of plagiarism-detection software in higher education reduce plagiarism?". Studies in Higher Education.* ***36*** *(7): 749–761. doi:10.1080/03075079.2010.523457.*
12. *Hoad, Timothy; Zobel, Justin (2003), "Methods for Identifying Versioned and Plagiarised Documents" (PDF), Journal of the American Society for Information Science and Technology,* ***54*** *(3): 203–215, CiteSeerX 10.1.1.18.2680, doi:10.1002/asi.10170*
13. *Stein, Benno (July 2005), "Fuzzy-Fingerprints for Text-Based Information Retrieval", Proceedings of the I-KNOW '05, 5th International Conference on Knowledge Management, Graz, Austria (PDF), Springer, Know-Center, pp. 572–579*
14. *Brin, Sergey; Davis, James; Garcia-Molina, Hector (1995), "Copy Detection Mechanisms for Digital Documents", Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data (PDF), ACM, pp. 398–409, doi:10.1145/223784.223855, ISBN 1-59593-060-4*
15. *Monostori, Krisztián; Zaslavsky, Arkady; Schmidt, Heinz (2000), "Document Overlap Detection System for Distributed Digital Libraries", Proceedings of the fifth ACM conference on Digital libraries (PDF), ACM, pp. 226–227, doi:10.1145/336597.336667, ISBN 1-58113-231-X, archived from the original (PDF) on 15 April 2012, retrieved 7 October 2011*
16. *Baker, Brenda S. (February 1993), On Finding Duplication in Strings and Software (Technical Report), AT&T Bell Laboratories, NJ, archived from the original (gs) on 30 October 2007*
17. *Khmelev, Dmitry V.; Teahan, William J. (2003), "A Repetition Based Measure for Verification of Text Collections and for Text Categorization", SIGIR'03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval, ACM, pp. 104–110, CiteSeerX 10.1.1.9.6155, doi:10.1145/860435.860456*
18. *Si, Antonio; Leong, Hong Va; Lau, Rynson W. H. (1997), "CHECK: A Document Plagiarism Detection System", SAC '97: Proceedings of the 1997 ACM symposium on Applied computing (PDF), ACM, pp. 70–77, doi:10.1145/331697.335176, ISBN 0-89791-850-9*
19. *Dreher, Heinz (2007), "Automatic Conceptual Analysis for Plagiarism Detection" (PDF), Information and Beyond: The Journal of Issues in Informing Science and Information Technology,* ***4****: 601–614*
20. *Muhr, Markus; Zechner, Mario; Kern, Roman; Granitzer, Michael (2009), "External and Intrinsic Plagiarism Detection Using Vector Space Models", PAN09 - 3rd Workshop on Uncovering Plagiarism, Authorship and Social Software Misuse and 1st International Competition on Plagiarism Detection (PDF), CEUR Workshop Proceedings,* ***502****, pp. 47–55, ISSN 1613-0073, archived from the original (PDF) on 2 April 2012*
21. *Gipp, Bela (2014), Citation-based Plagiarism Detection, Springer Vieweg Research, ISBN 978-3-658-06393-1*
22. *Gipp, Bela; Beel, Jöran (June 2010), "Citation Based Plagiarism Detection - A New Approach to Identifying Plagiarized Work Language Independently", Proceedings of the 21st ACM Conference on Hypertext and Hypermedia (HT'10) (PDF), ACM, pp. 273–274, doi:10.1145/1810617.1810671, ISBN 978-1-4503-0041-4*

23. *Gipp, Bela; Meuschke, Norman; Breitinger, Corinna; Lipinski, Mario; Nürnberger, Andreas (28 July 2013), "Demonstration of Citation Pattern Analysis for Plagiarism Detection", Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval (PDF), ACM, doi:10.1145/2484028.2484214*
24. *Gipp, Bela; Meuschke, Norman (September 2011), "Citation Pattern Matching Algorithms for Citation-based Plagiarism Detection: Greedy Citation Tiling, Citation Chunking and Longest Common Citation Sequence", Proceedings of the 11th ACM Symposium on Document Engineering (DocEng2011) (PDF), ACM, pp. 249–258, doi:10.1145/2034691.2034741, ISBN 978-1-4503-0863-2*
25. *Gipp, Bela; Meuschke, Norman; Beel, Jöran (June 2011), "Comparative Evaluation of Text- and Citation-based Plagiarism Detection Approaches using GuttenPlag", Proceedings of 11th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL'11) (PDF), ACM, pp. 255–258, doi:10.1145/1998076.1998124, ISBN 978-1-4503-0744-4*
26. *Gipp, Bela; Beel, Jöran (July 2009), "Citation Proximity Analysis (CPA) - A new approach for identifying related work based on Co-Citation Analysis", Proceedings of the 12th International Conference on Scientometrics and Informetrics (ISSI'09) (PDF), International Society for Scientometrics and Informetrics, pp. 571–575, ISSN 2175-1935*
27. *Holmes, David I. (1998), "The Evolution of Stylometry in Humanities Scholarship", Literary and Linguistic Computing, **13** (3): 111–117, doi:10.1093/llc/13.3.111*
28. *Juola, Patrick (2006), "Authorship Attribution" (PDF), Foundations and Trends Information Retrieval, **1**: 233–334, doi:10.1561/1500000005, ISSN 1554-0669*
29. *Portal Plagiat - Softwaretest 2004 (in German), HTW University of Applied Sciences Berlin, retrieved 6 October 2011*
30. *Portal Plagiat - Softwaretest 2008 (in German), HTW University of Applied Sciences Berlin, retrieved 6 October 2011*
31. *Portal Plagiat - Softwaretest 2010 (in German), HTW University of Applied Sciences Berlin, retrieved 6 October 2011*
32. *Potthast, Martin; Barrón-Cedeño, Alberto; Eiselt, Andreas; Stein, Benno; Rosso, Paolo (2010), "Overview of the 2nd International Competition on Plagiarism Detection", Notebook Papers of CLEF 2010 LABs and Workshops, 22–23 September, Padua, Italy (PDF)*
33. *Potthast, Martin; Eiselt, Andreas; Barrón-Cedeño, Alberto; Stein, Benno; Rosso, Paolo (2011), "Overview of the 3rd International Competition on Plagiarism Detection", Notebook Papers of CLEF 2011 LABs and Workshops, 19–22 September, Amsterdam, Netherlands (PDF)*
34. *Stein, Benno; Lipka, Nedim; Prettenhofer, Peter (2011), "Intrinsic Plagiarism Analysis" (PDF), Language Resources and Evaluation, **45** (1): 63–82, doi:10.1007/s10579-010-9115-y, ISSN 1574-020X*
35. *Potthast, Martin; Barrón-Cedeño, Alberto; Stein, Benno; Rosso, Paolo (2011), "Cross-Language Plagiarism Detection" (PDF), Language Resources and Evaluation, **45** (1): 45–62, doi:10.1007/s10579-009-9114-z, ISSN 1574-020X*
36. *Weber-Wulff, Debora (June 2008), "On the Utility of Plagiarism Detection Software", In Proceedings of the 3rd International Plagiarism Conference, Newcastle Upon Tyne (PDF)*
37. *"Plagiarism Prevention and Detection - On-line Resources on Source Code Plagiarism" Archived 15 November 2012 at the Wayback Machine.. Higher Education Academy, University of Ulster.*

38. Roy, Chanchal Kumar;Cordy, James R. (26 September 2007)."A Survey on Software Clone Detection Research". School of Computing, Queen's University, Canada.