Ako ťažké je obísť systém na odhaľovanie plagiátov?

Ján Genči
Technická univerzita v Košiciach
genci@tuke.sk

Abstrakt

V minulom školskom roku bol do života slovenských vysokých škôl zavedený centrálny systém na detekciu plagiátov. Cieľom jeho nasadenia je, ak nie zamedzenie, tak aspoň obmedzenie úrovne plagiátorstva v záverečných prácach absolventov univerzít. Podobné systémy našli uplatnenie oveľa skôr aj v iných štátoch, či už na komerčnom základe alebo na pôde samotných vzdelávacích inštitúcií. Podrobnosti o princípoch fungovania takýchto systémov bývajú často utajované a preto je zvyčajne ťažké zistiť, ako uvedené systémy fungujú. Pri skúmaní tejto otázky je ale možné urobiť záver, že hlavný dôraz sa v súčasnosti venuje algoritmom na detekciu plagiátov. Príspevok prezentuje prístupy, ktoré umožňujú pozmeniť obsah dokumentu - plagiátu tak, že vizuálne sa obsah dokumentu nezmení, ale pre potreby detekcie plagiátov dokument vykazuje iné parametre. Plagiát sa tak stáva neodhaliteľným.

Úvod

Prienik internetu do života spoločnosti nenecháva bokom ani otázky vzdelávania. Internet sa stal neoceniteľným prostriedkom pre zefektívnenie a sproduktívnenie štúdia tým, že umožňuje poskytovať vzdelávací obsah širokému spektru populácie (metódou kdekoľvek, ktokoľvek, kedykoľvek) nielen klasickými, ale aj alternatívnymi formami jeho prezentácie. Na druhej strane sa však stal aj miestom, kde si študenti zvykli jednoduchým spôsobom nájsť potrebné zdroje a tieto prezentovať bez adekvátneho „myšlienkového“ spracovania buď v takej istej forme alebo len s malými úpravami. Rozvoj sociálnych sietí priniesol do vzdelávania „ďalší rozmer“, spočívajúci v enormne rýchlej výmene informácií. Študenti v ďaleko väčšej miere, ako to bolo v minulosti, zdieľajú nielen vzdelávanie zdroje, ale aj výsledky svojej práce – či už ide o výsledky testov alebo nimi vypracované práce. Okrem verejne dostupných zdrojov, ako napr. stránky http://diplomovky.sme.sk a http://www.referaty.sk, študenti si organizujú vlastné diskusné fóra (viď napr. http://www.tu-ke.com) a dokonca chránené FTP servery pre výmenu rôznych typov zdrojov.

Odpisovanie, ktoré v našom vzdelávacom priestore (a pravdepodobne nielen vo vzdelávacom) existovalo desaťročia, prerástlo do ohromných rozmerov. Paradoxne však, práve internet, ako prostriedok efektívnej komunikácie, sa stal prostriedkom pre odhaľovanie plagiátov. Kým donedávna sa na školách plagiátorstvo viac-menej ticho tolerovalo a len najmarkantnejšie zistené prípady sa nejakým spôsobom síce postihovali, ale nemedializovali, v posledných rokoch len na Slovensku média odhalili niekoľko prípadov plagiátorstva aj v radoch učiteľov a odborníkov z praxe. Diskusie čitateľov k relevantným článkom svedčia o tom, aký je tento problém akútny.

Potreba riešenia problémov spojených s narastajúcim plagiarizmom viedla zodpovedné inštitúcie k zavádzaniu antiplagiátorských opatrení. Tie spočívajú na jednej strane v osvete ohľadom problematiky plagiarizmu predovšetkým v radoch študentov, na druhej strane zavádzajú technologické opatrenia vo forme systémov na odhaľovanie plagiátov.

I keď v našom priestore sa niekedy zdôrazňuje „iná kultúra“ vo vzťahu k plagiátorstvu v anglosaských krajinách, je zaujímavosťou, že práve odtiaľ prišli prvé aktivity týkajúce sa nasadzovania systémov pre odhaľovanie plagiátov. Asi najznámejším sú komerčné riešenia TutnItIn, resp. http://www.plagiarismdetect.com. Existuje však veľké množstvo ďalších riešení. V našom priestore je to systém THESES.CZ, vyvinutý na Masarykovej univerzite v Brne, systém Strikeplagiarism.com z Varšavy, či najnovšie systém obstaraný MŠ SR.

Detekcia plagiátov

Podľa [1] „Plagiátorstvo je nedovolené používanie cudzích publikovaných i nepublikovaných myšlienok, formulácií, poznatkov, výsledkov bádania alebo iných výsledkov tvorivej práce, ako aj ilustrácií, tabuliek, fotografií a pod. bez referencie“. Z uvedenej definície vyplýva, že plagiátorstvo môže mať veľmi veľa podôb – počnúc jednoduchým odkopírovaním textu, cez jeho mierne modifikácie nahradením niektorých slov adekvátnymi synonymami až po parafrázovanie textu. Miera modifikácie textu pritom nepriamoúmerne zodpovedá možnostiam technologickej detekcie plagiátu. Predpokladom technologického prístupu je, že plagiátor sa dopúšťa plagiarizmu predovšetkým preto, že nie je ochotný investovať čas do spracovania zadanej problematiky. V tomto prípade môžeme predpokladať, že plagiátor nebude mať záujem investovať do „prepracovania“ odpisovaného textu väčšie množstvo času, pretože v takom prípade by sa mu viac oplatilo plagiarizmu sa vzdať a spracovať originálny text.

Nemôžeme však vynechať aj prípadnú otázku neschopnosti plagiátora problematiku spracovať. V tomto prípade už plagiátor pravdepodobne nebude váhať do zmeny pôvodného textu investovať ďaleko väčšie množstvo času a prípadného zvyšku jeho invencie, aby zabránil odhaleniu plagiátu.

Na detekciu plagiátov bol navrhnutý celý rad algoritmov ([2], [5], [3], detailne rozobratých napr. v [6], [5], [10], prehľadovo napr. v [9]). Tieto algoritmy, na základe typov modifikácie textu, môžeme rozdeliť do niekoľkých základných kategórií:

1. algoritmy presnej zhody (exact match) – identifikujú v porovnávaných dokumentoch všetky zhodné reťazce maximálnej dĺžky;
2. algoritmy detekcie podobnosti (similarity search) - identifikujú v porovnávaných dokumentoch podobné reťazce, pričom miera podobnosti môže byť parametricky nastavená;
3. algoritmy porovnávajúce odvodené charakteristiky textu – napr. odtlačky (fingerprints) podtextov, štatistické rozdelenie n-gramov.

Spomenuté algoritmy sa spravidla aplikujú na tzv. plain-text verzie textov, ktoré vzniknú transformáciou pôvodného textu z niektorého z dokumentových formátov (napr. PDF, MS Office, Open Office a iné). Práce sú však autormi prezentované v pôvodných dokumentových formátoch a ich prípadná snaha o znemožnenie identifikácie plagiátov prebieha na tejto úrovni. Ich cieľom je znemožniť stotožnenie odpísaných častí pomocou aplikovaných algoritmov. Vzhľadom na to, že autori prác sú informovaní o technologickom spracovaní ich textov za účelom odhaľovania plagiátov, ich snahu o znemožnenie detekcie odpísaných častí textu budeme nazývať útokmi na algoritmy detekcie plagiátov aplikovaných v systémoch na ich odhaľovanie.

Útoky na algoritmy detekcie plagiátov

Ako bolo spomenuté vyššie, cieľom útoku je modifikovať odpísaný text v dokumente spôsobom, ktorý znemožní jeho odhalenie používanými algoritmami detekcie, ale zároveň zachová jeho pôvodnú formu. Navyše, vyžaduje sa, aby modifikácie boli relatívne nenáročné, ináč by pre útočníka strácali zmysel.

Algoritmy presnej zhody ([6], [8]) sú najzraniteľnejšími algoritmami s pohľadu možných útokov. Ako sme uviedli, cieľom týchto algoritmov je identifikovať zhodné reťazce maximálnej dĺžky. Avšak tieto algoritmy musia mať nastavenú aj spodnú hranicu detegovanej dĺžky. V opačnom prípade by algoritmus poskytol výstup zahltený množstvom krátkych reťazcov, pozostávajúcich zo spoločných koreňov slov, spoločných prefixov a sufixov, slov samotných resp. často používaných fráz - kombinácií slov. Pre útoky na tieto algoritmy stačí zabezpečiť zmenu jediného znaku v tzv. porovnávanom „okne“ textu. Možností, ako to dosiahnuť, bolo nami identifikovaných niekoľko:

E1. náhrada znaku medzera niektorým z iných znakov v znakovej sade vyzerajúcich ako medzera, ale v textovom súbore produkujúcom kód znaku rozdielny od znaku medzera;
E2. náhrada znaku medzera ľubovoľným iným znakom tak, aby tento znak splynul s pozadím textu;
E3. vložením nezobraziteľných znakov na ľubovoľné pozície textu;
E4. náhrada znakov jednej znakovej podsady podobne vyzerajúcimi znakmi inej znakovej podsady;
E5. náhrada niektorých znakov kombináciou viacerých znakov (napr. náhrada znaku á kombináciou znaku „a“ a znaku dĺžeň);

Útok E1 je možné dosiahnuť identifikovaním znakov podobných medzere v niektorom (alebo niektorých) z typov písma. Napr. v textovom editore MS Word v štandardne inštalovanom systéme Windows XP SK so systémom MS Office 2003 CZ, sú v písme *Arial Black* takto zadefinované znaky sady UNICODE-16 s kódmi F000-F008. V pomerne krátkom čase je možné identifikovať množstvo ďalších znakových sád obsahujúcich takéto znaky. Vizuálne je takýto text prakticky identický s pôvodným textom, z hľadiska kódov znakov to však neplatí.

Útok E2 spočíva v zmene vybraných znakov medzery na ľubovoľný iný znak tak, aby farba tohto znaku bola identická s farbou pozadia znaku. Tento útok sa môže zdať byť ľahko identifikovateľný pri konverzii pôvodného dokumentu do textového formátu, porovnaním kódov farieb písma a pozadia jednotlivých znakov. Avšak používaný 24-bitový RGB model umožňuje špecifikovať jemné rozdiely vo farbe pozadia a farbe písma, ktoré môžu byť voľným okom takmer nerozlíšiteľné – napr. pozadie (255,255,255), písmo (254,254,255).

Útok E3 je založený na skutočnosti, že niektoré znaky znakovej sady sú znakmi nezobraziteľnými. Napr. v písme označenom ako *normální text* pre písmo *Times New Roman* sú UNICODE-16 znaky s kódmi 202A-202E v texte nezobraziteľné. V plain-textovej reprezentácii sa však tieto znaky objavia s im korešpondujúcimi kódmi.

Útok E4 využíva skutočnosť, že niektoré znaky sú obsiahnuté vo viacerých znakových podsadách. Nám sa podarilo identifikovať, že súbor znakov *aeijosy* je obsiahnutý nielen v podsade Štandardná latinka, ale aj v podsade Cyrilika. Vzhľadom na to, že tieto znaky sú súčasťou toho istého typu písma, sú vizuálne prakticky nerozlíšiteľné. Samozrejme, im zodpovedajúce kódy znakov sú odlišné.

Útok E5 je umožnený spracovaním niektorých diakritických znamienok v znakovej sade. Konkrétne v znakovej sade *normální text* je pod kódom 0301 definovaný znak pre dĺžeň. Kombináciou niektorého z relevantných písmen a tohto znaku, dostaneme v texte zodpovedajúce dlhé diakritické písmeno, ktoré je však v plain-texte reprezentované dvojicou znakov.

Útoky E1-E5 je, samozrejme, možné v ľubovoľnej miere kombinovať.

Algoritmy detekcie podobnosti ([8], [3]) reťazcov sú spravidla založené na ohodnotení tzv. editačnej vzdialenosti – počtu operácií mazania, vloženia a zámeny znakov, ktoré je potrebné vykonať, aby sme jeden reťazec znakov transformovali na reťazec druhý. Všetky útoky voči algoritmom presnej zhody opísané ako E1-E5 je možné využiť aj v prípade útokov na algoritmy detekcie podobnosti. Na rozdiel od predchádzajúceho prípadu, kde stačilo vystihnúť/odhadnúť šírku okna, v ktorom sa porovnáva zhoda a v tomto okne zabezpečiť jedinú zmenu, v prípade algoritmov detekcie podobnosti je potrebné aplikovať niektorý z útokov E1-E5 (alebo ich kombináciu) tak, aby sa editačná vzdialenosť porovnávaných reťazcov „spoľahlivo“ dostala nad predpokladanú editačnú vzdialenosť. Algoritmy detekcie podobnosti, samozrejme, sú schopné identifikovať aj presnú zhodu reťazcov (editačná vzdialenosť 0).

Ďalším spôsobom v tejto kategórii je identifikácia tzv. najdlhšej spoločnej sekvencie znakov (Longest common subsequence, napr. [3]). Podobne ako v útokoch na algoritmy presnej zhody, v tomto prípade stačí zabezpečiť požadovanú šírku okna.

Algoritmy porovnávajúce odvodené charakteristiky textu ([8], [10]) predstavujú iný prístup k identifikácií zhodných, resp. podobných častí testu. Existuje niekoľko prístupov v tejto oblasti – spomeňme využitie odtlačkov (fingerprintov) alebo n-gramov.

Princíp identifikácie podobných reťazcov v porovnávaných textoch na základe odtlačkov spočíva vo vyčíslení odtlačku (fingerprint) časti spracovávaného reťazca. Tento prístup zabezpečuje efektívnu identifikáciu zhodných reťazcov. Postupne sa určujú odtlačky pre prekrývajúce sa časti textu. Pre určenie podobnosti nie je potrebné porovnávať celé reťazce, ale iba mieru zhody odtlačkov pre porovnávané texty. Reťazce pred vyčíslením odtlačku je možné normalizovať a tak zabezpečiť určenie podobných častí textu. Algoritmy založené na tomto princípe je možné zaradiť aj do prechádzajúcej skupiny - algoritmy detekcie podobnosti.

Útok na tento algoritmus je opäť možné realizovať aplikáciou útokov E1-E5 a/alebo ich kombináciou. Môžeme sa zaoberať otázkou odhadu dĺžky reťazca, na základe ktorého sa vypočítava odtlačok alebo na tento odhad rezignujeme a zmeny v texte opísané v útokoch E1-E5 aplikujeme na každé slovo.

n-gram je postupnosť *n* znakov textu. Číslo *n* nebýva vysoké, spravidla *n* nie je väčšie ako 5 (pri *n=1*, hovoríme o unigramoch, pri *n=2* o bigramoch, pri *n=3* o trigramoch atď.). Postupnosť n-gramov spolu s frekvenciou ich výskytu formuje tzv. profil dokumentu. Cavnar v [7] prezentoval, že prvých cca 300 najfrekventovanejších n-gramov relatívne spoľahlivo charakterizuje jazyk dokumentu. Frekvencia výskytu n-gramov za touto hranicou, podľa [7], charakterizuje viac obsah dokumentu. Práve túto časť profilu je možné použiť na klasifikáciu dokumentov podľa ich obsahu. Uvedený prístup už niekoľko rokov úspešne používame na odhaľovanie plagiátov zdrojových textov, v rámci nášho systému na odovzdávanie zadaní.

Nedostatkom „n-gramového“ prístupu je však to, že n-gramový profil charakterizuje dokument ako celok. Pokiaľ by sme chceli identifikovať plagiáty iba v častiach textu (čo je najčastejší prípad), bolo by potrebné spracovávať n-gramové profily pre jednotlivé časti textov (napr. odstavce, kapitoly) a následne tieto profily porovnávať. V takomto prípade vznikajú dva problémy:

1. pomerne veľká výpočtová náročnosť tohto procesu
2. „reprezentatívnosť“ n-gramového profilu pre pomerne krátke úseky textu.

Z uvedených dôvodov sme presvedčení, že opísaný prístup má význam iba na určovanie podobnosti väčších úsekov textu – celé práce alebo aspoň kapitoly.

Útok na algoritmus určenia podobnosti na báze n-gramového profilu znamená pokus o zmenu profilu samotného. Tento prístup je oproti útokom E1-E5 náročnejší, pretože ovplyvnenie n-gramového profilu si vyžaduje nielen vloženie väčšieho počtu znakov do pôvodného textu, ale aj text musí byť vyberaný cielene tak, aby ovplyvnil zmenu profilu želaným spôsobom. Ovplyvnenie aktuálneho profilu si vyžaduje jeho určenie, čo pre mnohých autorov môže byť problém, avšak pre študentov IT zamerania je to pomerne jednoduchá úloha. Po určení aktuálneho profilu je potrebné vybrať náhodné n-gramy, priradiť im frekvenciu výskytu na úrovni (alebo dokonca nad úrovňou) najfrekventovanejších n-gramov a sformovať relevantný, možno aj nezmyslený, text.

Ďalším krokom útoku je vloženie nového, vizuálne neidentifikovateľného textu do textu pôvodného. Prvou možnosťou je využitie útoku E2 s tým rozdielom, že na miesto znaku medzera vložíme nie jeden znak, ale celú sériu znakov. Elimináciu neprirodzenej šírky medzery, ktorá takýmto spôsobom vznikne, vykonáme zmenšením veľkosti písma (spravidla veľkosť 1, čo nám poskytne priestor pre väčší počet vkladaných znakov). Formátovaný text však obsahuje celý rad ďalších miest, kam je možné vložiť ďaleko väčší počet znakov, resp. slov, sformovaných zo zvolených n-gramov. Napr.:

- konce odstavcov;
- medzery medzi odstavcami;
- konce riadkov pri formátovaní „bez zarovnania vpravo“;
- priestor medzi riadkami pri riadkovaní väčšom ako 1;
- priestor okolo obrázkov

Takýmto spôsobom je možné napr. do priestoru medzi riadkami uložiť až okolo 1000 znakov, čo môže predstavovať asi 12 riadkov skrytého textu. Je zrejmé, že to dáva enormné možnosti, ako ovplyvniť n-gramový profil.

Vyššie opísaný n-gramový prístup je založený na určovaní n-gramových profilov na báze postupnosti znakov. Môžeme sa stretnúť s pojmom n-gram, založenom na postupnosti slov. V tomto prípade sa jednotlivé slová (alebo ich korene, resp. normalizované tvary – stem, lema) zvyčajne považujú za prvky n-gramu (abeceda). Spracovanie je, v tomto prípade, prakticky zhodné s prístupom založenom na odtlačkoch, len „okno“, nad ktorým sa vypočítavajú odtlačky nie je určené počtom znakov, ale počtom slov (teda počet znakov v jednotlivých oknách sa môže líšiť). Útok na tento algoritmus možno pohodlne vykonať útokmi E3-E5.

Praktické skúsenosti

Niektoré z opísaných útokov sme otestovali na systémoch Odevzdej.cz a systéme Turnitin. Pre účely testovania systému Odevzdej.cz sme použili dokument uložený v systéme dostupný na Theses.cz, aby sme mali istotu, že voči danému dokumentu bude nami odoslaný dokument testovaný. Výsledky percentuálnej zhody reportované systémom sú uvedené v tabuľke 1. Stĺpec Report v Tab.1 indikuje, že k protokolu o zhode bol pripojený výpis časti textu, ktorý bol označený ako podozrivý.

Tab. 1 Reportovaná zhoda pre niektoré typy útokov pre systém Odevzdej.cz

Útok	Report	Zhoda
Zámena znakov s dĺžňom na postupnosť dvoch znakov	Áno	17%
Zámena niektorých znakov medzera za postupnosti „ ak “ alebo „ ii “	Áno	8%
Zámena znaku „a“ za znak „a“ zo znakovej subsady cyrilika	Áno	9%
Zámena znakov „aeoc“ za znaky „aeoc“ zo znakovej subsady cyrilika	Nie	9%
Zámena znakov „aeo“ za znaky „aeo“ zo znakovej subsady cyrilika	Nie	0%
Vloženie nezobraziteľných znakov	Nie	0%

Pre otestovanie systému Turnitin sme použili časť článku dostupného na stránke encyklopédie Wikipedia. Typy útokov a výsledky sú uvedené v Tab. 2.

Tab. 2 Reportovaná zhoda pre niektoré typy útokov pre systém Turnitin

Útok	Report	Zhoda
Zámena medzery za takmer biele „n“ a biele „ or“ veľkosťou 1	Áno	0%
Vloženie nezobraziteľných znakov do slov	Áno	10%
Zámena znaku „a“ za znak „a“ zo znakovej subsady cyrilika	Áno	99%
Zámena znakov „aeoc“ za znaky „aeoc“ zo znakovej subsady cyrilika	Áno	99%
Zámena znakov „aeo“ za znaky „aeo“ zo znakovej subsady cyrilika	Áno	99%

Záver

Systémy na detekciu plagiátov sú nepopierateľne vhodným technologickým prostriedkom pre elimináciu plagiarismu, alebo aspoň zníženie jeho miery. A hoci v oblasti rozpracovania algoritmov pre detekciu plagiátov bol urobený obrovský kus práce, príspevok poukázal na prístupy, ktoré môžu ovplyvniť proces detekcie tak, že identifikácia podobných dokumentov môže byť sťažená, alebo, dokonca, znemožnená.

I napriek tomu, že z „reportov“ generovaných systémami, pokiaľ boli poskytnuté, bol ten-ktorý útok zvyčajne pomerne jednoducho identifikovateľný, čaká nás pravdepodobne ešte veľké množstvo práce v oblasti rozpracovania spôsobov identifikácie prezentovaných útokov. Niektoré z útokov je možné eliminovať pomerne jednoducho na úrovni algoritmov identifikácie plagiátov, niektoré si však vyžiadajú pomerne zložitú a náročnú analýzu v procese konverzie textu. Navyše, tu prezentovaný zoznam útokov nie je pravdepodobne ani zďaleka kompletný a v budúcnosti bude potrebné problematiku naďalej sledovať.

Nezaujímavým by zrejme bolo aj preskúmanie súčasných archívov dostupných textov, či sa niektoré z tu opísaných útokov už nevyskytli, resp. sofistikovanejšia inšpekcia textov pre identifikáciu ďalších typov útokov.

Referencie

[1] Dušan Meško: Ako vysvetliť študentom podstatu plagiátorstva : pohľad skúseného učiteľa. Zborník Ako kvalitne učiť? Skúsenosti začínajúcich VŠ učiteľov. Bratislava, 2007.

[2] Karp R. M., Rabin M. O.: Efficient randomized pattern-matching algorithms. IBM Journal of Research and Development archive, Volume 31 , Issue 2, 1987, Pp: 249 – 260, 1987.

[3] Smith, T. F., Waterman, M. S.: Identification of common molecular subsequences. In: Journal of Molecular Biology, 147:195-197, 1981.

[4] Schleimer S., Wilkerson D. S., Suken, A.: Winnowing: local algorithms for document fingerprinting. In Proceedings of ACM SIGMOD Int. Conf. on Management of Data, 2003.

[5] Charras Ch., Lecroq T.: Handbook of Exact String Matching Algorithms. King's College London Publications, 2004.

[6] Crochemore M., Hancart Ch., Lecroq T.: Algorithms on Strings. Université de Rouen, 2001.

[7] Cavnar W. B., Trenkle J. M.: N-gram-based text categorization. Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval, 1994.

[8] Gusfield D.: Algorithms on Strings, Trees and Sequences. Cambridge University Press, 1997.

[9] Pločica O., Telepovská H.: Metódy detekcie plagiátorstva. Konferencia UNINFOS 2009, Nitra, 2009.

[10] Mozgovoy, M.: Enhancing Computer-Aided Plagiarism Detection, Dissertation, University of Joensuu, 2007.

How difficult is it to overcome plagiarism detection system?

Ján Genči
Technická univerzita v Košiciach
genci@tuke.sk

Abstract

In the previous academic year Slovak universities has started to use a central system to detect plagiarism. The aim of the deployment is, if not prevent, then at least limit the level of plagiarism in theses of university graduates. Similar systems have found application much earlier in other countries, implemented either on a commercial basis or by educational institutions themselves. The principles of operation of such systems are often hidden for its users and therefore it is usually difficult to see how the system works. During the examination of the systems it can be concluded that the main attention is given to the design and implementation of algorithms for plagiarism detection. The paper presents approaches whose allow to manipulate the content of the plagiarized document in such way, that the visual content of the document will not change, but for the purpose of detecting plagiarism document has other parameters. Process of plagiarism detection becomes inefficient.