StuCoSReC

Proceedings
of the 2018
5th Student
Computer
Science
Research
Conference

StuCoSReC

Proceedings of the 2018 5th Student Computer Science Research Conference

Edited by

Iztok Fister jr. and Andrej Brodnik

Reviewers and Programme Committee

Klemen Berkovič • University of Maribor, Slovenia
Lucija Brezočnik • University of Maribor, Slovenia
Patricio Bulić • University of Ljubljana, Slovenia
Zoran Bosnić • University of Ljubljana, Slovenia
Borko Bošković • University of Maribor, Slovenia
Janez Brest • University of Maribor, Slovenia
Andrej Brodnik • University of Primorska and University
of Ljubljana, Slovenia

Haruna Chiroma • Gombe State University, Nigeria
Mojca Ciglarič • University of Ljubljana, Slovenia
Jani Dugonik • University of Maribor, Slovenia
Iztok Fister • University of Maribor, Slovenia
Iztok Fister Jr. • University of Maribor, Slovenia
Matjaž Gams • Jozef Stefan Institute, Slovenia
Tao Gao • North China Electric Power University, China
Andres Iglesias • Universidad de Cantabria, Spain
Branko Kavšek • University of Primorska, Slovenia
Miklos Kresz • University of Szeged, Hungary
Niko Lukač • University of Maribor, Slovenia
Uroš Mlakar • University of Maribor, Slovenia
Marjan Mernik • University of Maribor, Slovenia
Eneko Osaba • University of Deusto, Spain
Vili Podgorelec • University of Maribor, Slovenia
Peter Rogelj • University of Primorska, Slovenia
Xin-She Yang • Middlesex University, United Kingdom
Damjan Vavpotič • University of Ljubljana, Slovenia
Grega Vrbančič • University of Maribor, Slovenia
Aleš Zamuda • University of Maribor, Slovenia
Nikolaj Zimic • University of Ljubljana, Slovenia
Borut Žalik • University of Maribor, Slovenia

Published by

University of Primorska Press
Titov trg 4, SI-6000 Koper

Editor-in-Chief

Jonatan Vinkler

Managing Editor

Alen Ježovnik

Koper, 2018

ISBN 978-961-7055-26-9 (pdf)

www.hippocampus.si/ISBN/978-961-7055-26-9.pdf

ISBN 978-961-7055-27-6 (html)

www.hippocampus.si/ISBN/978-961-7055-27-6/index.html

DOI: <https://doi.org/10.26493/978-961-7055-26-9>

© University of Primorska Press

Kataložni zapis o publikaciji (CIP) pripravili v Narodni in
univerzitetni knjižnici v Ljubljani

COBISS.SI-ID=296769024

ISBN 978-961-7055-26-9 (pdf)

ISBN 978-961-7055-27-6 (html)

Preface

It is no longer necessary to emphasize that computer science is omnipresent today. Terms as digitalization, Industry 4.0 etc. are frequently heard in the mass media and are becoming a part of our everyday life. The rapid advances in computer science require a large investment in research and development. In particular, it is important to educate young scientists, which is one of the objectives of the StuCoSReC conference.

In front of you is the proceedings of the fifth StuCoSReC conference. The StuCoSReC is intended for masters & PhD students to show their research achievements. Socializing is also important and the conference is an excellent opportunity for students to get to know each other. The conference connects students of computer science and goes beyond the invisible limits of faculties. The uniqueness of StuCoSReC conference is that it is organized each year by different higher education institution. The University of Ljubljana - Faculty of Computer and Information Science is proud to host the conference this year.

Twelve papers addressed this conference, covering several topics of the computer science. All the papers were reviewed by two international reviewers and accepted for the oral presentation.

ref. prof. dr. Nikolaj Zimic

Preface

II

Papers

<i>Regulacija plovca v zračnem tunelu (zračna levitacija) na osnovi mehke logike</i> • Primož Bencak, Dominik Sedonja, Andrej Picej, Alen Kolman, Matjaž Bogša	5–13
<i>Razpoznavanje literature v dokumentih in določanje njihovih sklicev v besedilu</i> • Matej Moravec, David Letonja, Jan Tovornik, Borko Boškovič	15–19
<i>Klasifikacija samomorilskih pisem</i> • Dejan Rupnik, Denis Ekart, Gregor Kovačevič, Dejan Orter, Alen Verk, Borko Boškovič	21–25
<i>Načrtovanje poprocesorja za pretvorbo NC/ISO G-kode v translacije za 3/5 osnega robota ACMA ABB XR701</i> • Gregor Germadnik, Timi Karner, Klemen Berkovič, Iztok Fister	27–33
<i>Using constrained exhaustive search vs. greedy heuristic search for classification rule learning</i> • Jamolbek Mattiev, Branko Kavšek	35–38
<i>Real-time visualization of 3D digital Earth</i> • Aljaž Jeromel	39–42
<i>Towards an enhanced inertial sensor-based step length estimation model</i> • Melanija Vežočnik, Matjaž Branko Jurič	43–46
<i>Breast Histopathology Image Clustering using Cuckoo Search Algorithm</i> • Krishna Gopal Dhal, Iztok Fister Jr., Arunita Das, Swarnajit Ray, Sanjoy Das	47–54
<i>Time series classification with Bag-Of-Words approach post-quantum cryptography</i> • Domen Kavran	55–59
<i>The problem of quantum computers in cryptography and post-quantum cryptography</i> • Dino Vlahek	61–64
<i>Named Entity Recognition and Classification using Artificial Neural Network</i> • Luka Bašek, Borko Boškovič	65–70
<i>Context-dependent chaining of heterogeneous data</i> • Rok Gomišček, Tomaž Curk	71

Razpoznavanje literature v dokumentih in določanje njihovih sklicev v besedilu

Matej MORAVEC

*Fakulteta za elektrotehniko,
računalništvo in informatiko
Koroška cesta 46,
2000 Maribor, Slovenija
matejmoravec19@gmail.com*

Jan TOVORNIK

*Fakulteta za elektrotehniko,
računalništvo in informatiko
Koroška cesta 46,
2000 Maribor, Slovenija
jan.tovornik@gmail.com*

David LETONJA

*Fakulteta za elektrotehniko,
računalništvo in informatiko
Koroška cesta 46,
2000 Maribor, Slovenija
david.letonja@gmail.com*

Borko BOŠKOVIČ

*Fakulteta za elektrotehniko,
računalništvo in informatiko
Koroška cesta 46,
2000 Maribor, Slovenija
borko.boskovic@um.si*

POVZETEK

Pojav plagiatorstva je vedno bolj pogost, saj v današnjem času obstaja zelo veliko elektronskih virov, preko katerih lahko dostopamo do želene vsebine. V ta namen je bilo razvitih veliko različnih pristopov za detekcijo plagiatov. V članku predstavimo pristop, kjer v dokumentih razpoznamo navedeno literaturo in znotraj dokumenta poiščemo, kje se nahajajo sklici v besedilu. Uspešnost našega pristopa smo preizkusili s pomočjo različnih metrik. Dosegli smo 67 % natančnost in 80 % mero F1 za razpoznavo virov.

Ključne besede

razpoznavanje literature, analiza citatov, plagiatorstvo, podobnost datotek

1. UVOD

Plagiat je delo, ki je prepisano, povzeto od drugod in objavljeno, prikazano kot lastno [3]. Večina strokovnih dokumentov vsebuje navedeno literaturo. Če se literatura nahaja v dokumentu ali ne, je seveda odvisno od vrste dokumenta.

Razpoznavanja literature in kasneje iskanja njihovih sklicev znotraj besedila smo se lotili z idejo, da bi na tak način pripomogli k večji natančnosti programov, ki preverjajo, ali je določeni dokument plagiat ali ne. Definicija plagiata v SSKJ pravi, da je plagiat tisto, "kar je prepisano, prevzeto od drugod in objavljeno, prikazano kot lastno, navadno v književnosti" [3].

Pristop razpoznavanja literature in iskanja njihovih sklicev

v dokumentih, predstavljen v tem članku, temelji na uporabi regularnih izrazov in posameznih korakov pregledovanja dokumentov, predstavljenih v naslednjih poglavjih. Glavna prednost našega pristopa je, da je neodvisen od jezika, v katerem je dokument napisan.

Članek je strukturiran tako, da v naslednjih poglavjih predstavimo sorodno delo, nato opišemo delovanje našega pristopa, opišemo rezultate eksperimenta in na koncu podamo kratek zaključek, kjer povzamemo naše ugotovitve skozi celotno delo.

2. SORODNA DELA

V članku [7] sta avtorja predstavila nov pristop k odkrivanju plagiatorstva, imenovan Analiza vrstnega reda citatov ali COA (Citation Order Analysis). Ta deluje na podlagi analize citatov in sklicevanja na vire oziroma literaturo ter vsebuje naslednje korake:

1. preoblikovanje dokumenta za procesiranje citatov in njihovih pojavitev v dokumentu,
2. ujemanje citatov z njihovimi navedbami v literaturi,
3. med dokumenti se preveri podobnost citatov. V osnovni različici sistema se upošteva samo vrstni red citatov, v naprednejši različici sistema pa se ocenjuje tudi razdalja med dvema citatoma. Če se dokument prevede v drugi jezik, se lahko vrstni red citatov znotraj stavkov ali odstavkov spremeni zaradi različnih struktur stavkov ali drugačnega načina pisanja.

Za ocenitev pristopa so opravili preizkuse na 800.000 prosto dostopnih znanstvenih publikacij, med katere so tudi skrili 20 posebej zasnovanih plagiariziranih dokumentov. Da je scenarij izgledal bolj realističen, so nekaj citatov izbrisali, dodali par novih, nekaterim spremenili stil, nekatere pa med seboj zamenjali. Pristop je od 20 testnih dokumentov uspešno odkril 19 in na stotine drugih dokumentov, ki so vsebo-

vali vsaj nekaj plagiriziranih odsekov. Več kot ima dokument navedenih citatov, težje je odkriti plagirizem. Preoblikovanje dokumentov iz formata pdf v xml je trajalo 2 sekundi za vsak dokument. V 96 % vseh zgoraj omenjenih dokumentov/publikacij je bilo ujemanje citatov z njihovimi navedbami v bibliografiji uspešno. Največja prednost opisanega pristopa je neodvisnost od jezika, v katerem je dokument napisan, in neodvisnost do parafraziranja. Časovna zahtevnost je v primerjavi s pristopi, ki temeljijo na podobnosti besedil, zanemarljiva. Največja slabost pristopa pa je odvisnost od pravilnega citiranja. Slabost tega pristopa je ob enem prednost pristopov, ki temeljijo na podobnosti besedil, zato se za doseganje najboljših rezultatov priporoča kombinirana uporaba obeh pristopov.

V članku [8] so še opisani rezultati, ki so jih avtorji pridobili ob avtomatski obdelavi ogromne količine znanstvenih člankov v skladu z natančno določenimi kriteriji ocenjevanja. Populacijo znanstvenih člankov, ki so jo uporabljali, so uredili v skupine tako, da so zadostili kriterijem medsebojne povezanosti, tj. povezanost v tematiki. Dokumenti iste tematike kažejo na zelo visoko stopnjo korelacije med seboj. Avtorji dokažejo, da lahko z njihovim pristopom zelo dobro določimo tematiko dokumenta le s pomočjo razpoznanne literature v dokumentih.

Delo Irene Marshakove [9] opisuje novo formalno metodo za klasifikacijo dokumentov. Ta metoda temelji na analizi referenc oziroma sklicev s pomočjo indeksiranja literature znanstvenih člankov. Vsakemu sklicu se dodeli indeks od začetka do konca dokumenta. Po dodelitvi indeksov dokumentom se le-ti indeksi in številka reference oziroma sklica medsebojno primerjajo. Primer, da imata dve ali več del veliko sklicev z enakimi indeksi, nakazuje, da je morebiti eno ali več del izmed teh plagiat.

Članek [4] opisuje, kako je potekalo prvo mednarodno tekmovanje v odkrivanju plagiatorstva. Na tem tekmovanju se je izvajala delavnica PAN - Delavnica o odkrivanju plagiatorstva, avtorstva in zlorabe socialne programske opreme (zloraba osebnih podatkov preko družabnih omrežij). To tekmovanje oziroma natečaj je bil razdeljen na več področij (13 različnih: različne težavnosti, več vrst nalog itd.) in to z enim razlogom, da bi preprečili kršitve, kot so plagiatorstvo in kraja osebnih podatkov. "Stranski efekt" tega tekmovanja je bil postaviti oceno, kako dobro detektiramo plagiatorstvo in varnost osebnih podatkov bodisi iz avtorskih del ali družabnih omrežij. Hkrati pa izbrati množico ljudi, ki bi razvijala različne algoritme za bolj varni jutri.

Članek [10] opisuje novo obliko združevanja dokumentov imenovano: so-citiranje. Je pristop, kjer ob navedbi vira navedemo skupaj več virov (dokumente, knjige, internetne vire, na katere se nato sklicujemo). Avtorji predpostavljajo uporabo so-citiranja zato, da zmanjšamo količino literature. Tako združujemo vire, ki temeljijo na isti tematiki, so iz iste založbe, imajo iste avtorje, ali pa preprosto so-citiramo zato, ker najdemo eno stvar iz dveh različnih virov.

3. PREDLAGANA METODA

Predlagana metoda vsebuje tri faze. V prvi fazi detektira poglavje, ki vsebuje navedeno literaturo. V drugi fazi zno-

traj poglavja razpozna posamezno literaturo in te oznake literature doda v seznam. V tretji fazi pregleda celoten dokument in poišče, na kateri strani in v katerem odstavku znotraj strani se nahajajo sklici na literaturo.

Preden smo pričeli z razpoznavanjem in gradnjo seznama literature, smo zaradi različno strukturiranih dokumentov naleteli na prvi izziv. Naša prva naloga je bila, da smo v dokumentu najprej našli, kje se nahaja poglavje z navedeno literaturo. To smo rešili tako, da smo zapisali tak regularni izraz, ki je razpoznaval čim več različnih načinov zapisa poglavja, ki vsebuje literaturo.

Vsebinsko poglavja z navedeno literaturo smo ravno tako preiskali s pomočjo regularnih izrazov. Z namenom razpoznavanja čim večjega števila virov znotraj poglavja smo zapisali večje število regularnih izrazov. Rešitve smo se lotili tako, da smo razpoznavali vrstico po vrstico znotraj poglavja z navedeno literaturo. Ob posameznem najdenem viru smo zapisali njegovo oznako (številko), v za to namenjen seznam. Ta seznam smo zgradili z namenom za kasnejše iskanje sklicev na posamezen vir v besedilu.

Množica dokumentov, ki smo jo uporabili kot učno množico za naš pristop je vsebovala strokovna oziroma znanstvena dela z različnih fakultet. Vsi dokumenti so bili zapisani v slovenskem jeziku in so po večini bili drugače strukturirani. Zaradi različno strukturiranih dokumentov je bilo razpoznavanje sklicev toliko večji izziv.

Za prepoznavanje poglavja, ki je vsebovalo navedeno literaturo, smo uporabili regularni izraz, specifičen za element paragraf (<p>). Ta regularni izraz lahko razbijemo na več delov:

- V prvem delu, poiščemo vse odstavke <p>, ki imajo vsebinsko atributa *xml:id*, ob pb (page break) in p (odstavku) celo številko.
- V drugem delu se prvemu regularnemu pravilu, pridružuje iskanje odstavkov <p> v poljubnem jeziku. V primeru, da bi želeli poiskati le določen jezik, bi morali vsebinsko atributa *xml:lang* specifično navesti.
- V zadnjem delu pa k prej navedenima praviloma dodamo pravilo za iskanje odstavkov <p> po specifičnih naslovih, pod katerimi se predstavlja literatura.

Regularni izraz poišče celoten odstavek, v katerem je podan naslov poglavja literature. Ker naslov poglavja najverjetneje vsebuje oštevilčenje, mora regularni izraz to tudi razpoznati. Upoštevati moramo tudi dejstvo, da obstaja več vrst oštevilčevanja, kar nam dodatno oteži razpoznavanje. Po zaporedni številki poglavja sledi naslov poglavja, pri katerem preverjamo naslednje besedne zveze (upoštevajo se velike in male črke):

- seznam virov,
- seznam literature,
- viri,

- literatura,
- seznam uporabljenih virov in
- reference.

Če vzorec ustreza določenemu izrazu, se najdeno poglavje tretira kot poglavje literature, ki vsebuje seznam literature oziroma virov.

Literatura je lahko podana na več načinov:

- oštevilčena z navadnimi oklepaji,
- oštevilčena z oglatimi oklepaji ali
- neoštevilčena.

Da smo razpoznali te tri načine, smo uporabili 2 različna regularna izraza, enega za oštevilčen način in enega za neoštevilčen način podajanja literature. Oba regularna izraza, za vsako podano literaturo, podata v seznam naslednje podatke:

- številka strani, na kateri je literatura podana,
- zaporedna številka v seznamu (če le-ta obstaja),
- prva beseda (kot priimek avtorja).

Ta seznam se v nadaljevanju uporabi za lažjo in bolj organizirano iskanje referenc.

S ciljem poiskati čim več sklicev v besedilu, smo zapisali različne regularne izraze. V našem pristopu smo zagotovili odkrivanje sklicev tam, kjer avtor dokumenta navede sklice na način, da v oglate ali navadne oklepaje zapiše posamezno oznako vira ali pa navede več virov znotraj oklepajev. Naš pristop najde tudi sklice, kjer avtor v oklepajih navede prvega avtorja.

Za pravilno prepoznavanje poglavij, literature/virov in sklicev smo uporabili regularne izraze. Za vsak korak v opisanem postopku se je uporabil svoj regularni izraz.

Za prepoznavanje referenc oziroma sklicev na literaturo smo uporabili kar 4 različne regularne izraze, ki razpoznavajo naslednje načine sklicevanja:

- enoštevilično v navadnih in oglatih oklepajih ((0), [0]),
- večštevilično v navadnih in oglatih oklepajih ((1, 2, 3), [1, 2, 3]),
- avtor brez oklepajev (Vir: avtor ...),
- avtor z navadnimi in oglatimi oklepaji ((avtor ...), [avtor ...]).

Iskanje in prepoznavanje referenc se opravlja nad celotnim dokumentom in z največ regularnimi izrazi, zato je ta korak v večini primerov časovno najzahtevnejši.

4. REZULTATI

V tem poglavju bomo primerjali dobljene rezultate z ročno dobljenimi rezultati in prikazali uspešnost našega pristopa.

Kontingenčna tabela je tip tabele, zapisan v matrični obliki, ki prikazuje frekvenčno porazdelitev več spremenljivk. Veliko se uporablja pri inženirskih, znanstvenih raziskavah ipd. Tabela prikaže osnovno idejo medsebojne povezave med dvema spremenljivkama in lahko pomaga pri iskanju medsebojne interakcije [1].

Tabela 1: Zapis kontingenčne tabele [5].

podatki / klasifikacija	pravičen sklic	napačen sklic
pravičen sklic	resnični pozitivni (tp)	lažno pozitivni (fp)
napačen sklic	lažno negativni (fn)	resnični negativni (tn)

Na Sliki 1 so prikazane vrednosti in njihove oznake, ki jih lahko klasificiramo iz podatkov. V našem primeru *lažno negativne* vrednosti predstavljajo sklice, ki jih je program našel, a v dokumentu ne obstajajo. *Resnično pozitivne* vrednosti predstavljajo sklice, ki v dokumentu dejansko obstajajo in jih je program našel, ter *lažno pozitivne* vrednosti so tiste, ki v dokumentu obstajajo, a jih program ni našel.

Slika 1: Prikaz vrednosti, ki jih lahko vrne klasifikator [2].

Kontingenčna matrika je za ocenjevanje uspešnosti našega pristopa uporabna, saj lahko s pomočjo le-te izračunamo

natančnost, preciznost, priklic in mero F1, kot prikazujejo naslednje enačbe:

$$accuracy = \frac{tp + tn}{tp + tn + fp + fn} \quad (1)$$

$$precision = \frac{tp}{tp + fp} \quad (2)$$

$$recall = \frac{tp}{tp + fn} \quad (3)$$

$$F1 = \frac{2PR}{P + R} \quad (4)$$

Za izračun uspešnosti smo uporabili 100 dokumentov. Od tega smo jih 31 izločili, saj so bili sklici navedeni tako, da niti človek ne bi znal razpoznati, na kateri vir se oseba v dokumentu sklicuje. Največji problem v dokumentih je torej predstavljalo naštevane virov tako, da sta bila pod isto številko naštet dva ali več virov. Zato smo za izračun uporabili 69 dokumentov in na koncu izračunali točnost, natančnost, priklic in metriko F1. Za izračun uspešnosti smo uporabili mikro povprečenje, kjer smo združili dobljene rezultate klasifikatorja.

V Tabeli 2 so prikazani rezultati posameznih izračunov.

Tabela 2: Metrike uspešnosti razpoznavanja literature.

Metrika	Rezultat
Točnost (angl. accuracy)	0,67
Natančnost (angl. precision)	0,88
Priklic (angl. recall)	0,74
Metrika F1 (angl. F1 score)	0,80

Uspešnost našega programa, ki smo jo izračunali na podlagi mikro povprečenja, je 67 %.

4.1 Eksperiment za ugotavljanje plagiatorstva

Z namenom prikaza uporabnosti našega pristopa v smislu ugotavljanja plagiatorstva med dokumenti smo izvedli en manjši eksperiment. Pri tem smo uporabili dva dokumenta. En dokument je predstavljal že objavljeno delo, drug dokument pa smo s tem dokumentom v smislu navajanja literature in navajanja sklicev primerjali. Navedeni potek primerjave smo izvedli trikrat (z različnimi dokumenti).

Pri prvi primerjavi dveh dokumentov, smo v objavljenem dokumentu z našo metodo našli 34 pravih sklicev in nič napačnih. V dokumentu za primerjavo z objavljenim pa smo našli 52 pravih sklicev in nič napačnih. Ugotovili smo, da je od teh 52 sklicev 34 takih, ki se pojavijo na isti strani, v istem odstavku kot v objavljenem dokumentu. Na podlagi teh rezultatov lahko z veliko verjetnostjo trdimo, da gre za plagiat, saj je zelo majhna verjetnost, da bi dva dokumenta imela na istih straneh, v istih odstavkih tako veliko sklicev na literaturo. V primerjavi drugih dveh dokumentov smo v objavljenem dokumentu našli 37 pravih sklicev, 2 sklica

pa nismo našli. V dokumentu za primerjavo pa smo našli le 20 pravih sklicev, a 5 sklicev nismo našli. Med vsemi sklici ni bilo nobenega, ki bi se nahajal na isti strani, v istem odstavku. Na podlagi takih rezultatov z našo metodo nismo ugotovili oziroma ne moremo ugotoviti, če gre za plagiat ali ne. Na podlagi rezultatov tretje izvedbe primerjave dveh dokumentov z našo metodo prav tako ne moremo ugotoviti, če gre za plagiat ali ne. Primerjave sploh ne moremo narediti, saj naša metoda od 6 sklicev v objavljenem dokumentu ni našla nobenega. V dokumentu za primerjavo je sicer našla 20 sklicev, vendar jih v tem primeru ne moremo uporabiti za primerjavo.

Na podlagi izvedenega eksperimenta lahko vidimo, v katerih primerih je predlagana metoda uporabna. Če so avtorji dokumentov dosledni in "pravilno" navajajo sklice in literaturo, potem lahko z uporabo predlagane metode dokaj učinkovito ugotovimo, ali je dokument plagiat ali ne, oziroma lahko v primeru uporabe ostalih metod za preverjanje plagiatorstva s to metodo povečamo oziroma zmanjšamo verjetnost, da je delo plagiat.

5. UGOTOVITVE IN PROBLEMI

Skozi razvoj našega pristopa k odkrivanju plagiatorstva smo naleteli na kar nekaj težav. Glavna težava v našem delu je bil način navajanja in sklicevanja na vire. Kot je zabeleženo v navodilih za pisanje zaključnih del [6], vidimo, da je navedenih kar nekaj napotkov, kako pravilno navajati in se nato sklicevati na vire.

Navedbo virov in literature je potrebno zapisati kot seznam na koncu zaključnega dela. V ta seznam je potrebno zabeležiti vse vire, na katere smo se sklicevali v zaključnem delu. Vsakega izmed virov je potrebno oštevilčiti z arabskimi številkami znotraj oglatih oklepajih (Slika 2).

[1] Avtor. Info. Povezava. Dostopano: 20. 7. 2018

Slika 2: Pravilna navedba vira.

Seznam literature in virov je potrebno urediti po abecednem vrstnem redu avtorjev. V primeru, da ni navedenih avtorjev, uredimo po naslovu vira. Pomanjkljivost in napačno sklicevanje virov zmanjšuje vrednost zaključnega dela [6].

Težave nam je povzročalo tudi iskanje navedenih sklicev v seznamu literature in virov. Ta seznam so nekateri študentje navajali napačno in na več različnih načinov:

1. naštevane z vezajem (-),
2. naštevane s piko (•),
3. seznam literature in virov NI bil naveden za vsemi poglavji na svoji strani, ampak je bil mnogokrat pred mnogimi poglavji, včasih kar na enaki strani kot neko poglavje.

Zaradi vseh teh napak naša programska rešitev ni mogla uspešno napolniti svojega seznama z viri, na podlagi katerih je nato iskala sklice po dokumentu. Ta problem je eden

izmed glavnih razlogov, zakaj smo morali izmed 100 dokumentov zavreči 31 dokumentov.

Razlog, kateremu se prav tako nismo morali izogniti, je bilo napačno sklicevanje virov. Avtorji so se na sam vir sklicevali napačno na več načinov:

1. sklicevanje brez oklepajev () ali brez zavitih oklepajev [], v katerih bi naj bila navedena zaporedna številka vira oziroma avtor,
2. sklicevanje v obliki 1. ali 1,
3. sklicevanje v obliki (vir1 vir2 vir3 itd.), kjer sklici med seboj niso bili pravilno ločeni. Za ločevanje sklicev so študentje uporabljali znake, kot je na primer podpičje (;),
4. sklicevanje na vir, ki na seznamu virov ne obstaja,
5. sklicevanje po začetnicah imena oziroma priimka avtorja,
6. nekateri študentje so se sklicevali na prilogo na enak način kot na vir, zato je naša programska rešitev napačno razpoznala sklic.

Probleme nam je povzročala tudi implementacija regularnih izrazov. Problem je bil ta, da so avtorji včasih navajali in se sklicevali na vire na dva ali več različnih načinov. Potrebno je bilo zapisati vgnezdene regularne izraze (glej poglavje 3), ki so v nekaterih dokumentih delovali, v nekaterih pa ne. Nekateri vgnezdene regularne izraze je bilo potrebno ločiti na več regularnih izrazov. Tako smo najprej preverili najpogostejše uporabljene načine navedbe in sklicevanja virov, nato pa tiste manj pogoste, da smo tako pokrili čim več načinov navedb.

6. ZAKLJUČEK

Plagiatorstvo je zaradi čedalje lažjega dostopa do elektronskih virov vse bolj pogosto. Skozi raziskovalno nalogo smo prikazali, kako bi znotraj dokumenta razpoznali literaturo in poiskali njihove sklice v besedilu. Vključitev takega načina preverjanja plagiarizma bi po našem mnenju veliko pripomogel k bolj natančni detekciji plagiata, saj veliko tistih, ki poizkusijo oddati plagiat, spremenijo vsebino, literatura pa ostane nespremenjena. Največja prednost našega pristopa je torej ta, da je pristop neodvisen od jezika, v katerem je dokument napisan.

Z uspešnostjo našega pristopa smo zadovoljni, vendar je uspešnost razpoznavne literature in kasneje določitve njihovih sklicev v besedilu v veliki meri odvisna od "pravilnosti" sestave dokumenta. Uspešnost bi lahko izboljšali tako, da bi zapisali še več bolj specifičnih regularnih izrazov in tako učinkoviteje razpoznali literaturo v različnih dokumentih.

7. LITERATURA

- [1] Kontingenčna tabela. https://en.wikipedia.org/wiki/Contingency_table. Dostopano 10. 5. 2018.
- [2] Metrika f1. https://en.wikipedia.org/wiki/F1_score. Dostopano 10. 5. 2018.
- [3] Plagiat. http://bos.zrc-sazu.si/cgi/a03.exe?name=sskj_testa&expression=ge%3Dplagiat*&hs=1. Dostopano 29. 4. 2018.
- [4] E. S. M. K. E. A. Benno Stein, Paolo Rosso. Pan workshop. uncovering plagiarism, authorship and social software misuse. <https://pdfs.semanticscholar.org/160b/400d726eb042d0867d537c447e858716e7b7.pdf>. Dostopano 8. 6. 2018.
- [5] J. D. Borko Boškovič and J. Brest. *Študijska literatura pri predmetu Jezikovne tehnologije*. 2018.
- [6] FERL. Navodila za pisanje zaključnih del na študijskih programih prve in druge stopnje um ferl. 2018.
- [7] B. Gipp and J. Beel. Citation based plagiarism detection - a new approach to identify plagiarized work language independently. *HT '10 Proceedings of the 21st ACM conference on Hypertext and hypermedia*, 2010.
- [8] M. M. Kessler. Bibliographic coupling between scientific papers. *American Documentation*, 1963.
- [9] I. V. Marshakova. System of document connections based on references. <http://garfield.library.upenn.edu/marshakova/marshakovanauchtechn1973.pdf>. Dostopano 8. 6. 2018.
- [10] H. Small. *Co-citation in the scientific literature: a new measure of the relationship between two documents*. 1973.

University of Primorska Press
www.hippocampus.si
ISBN 978-961-7055-26-9