ISSN 1648-8776
JAUNŲJŲ MOKSLININKŲ DARBAI. Nr. 47 (1). 2017 Doi: 10.21277/jmd.v47i1.88

DUOMENŲ IŠRINKIMO INTERNETO PUSLAPIUOSE ALGORITMAS, PAREMTAS DUOMENŲ TARPUSAVIO PANAŠUMU

Kiril Griazev[1], Simona Ramanauskaitė[2]
[1]Šiaulių universitetas, [2]Vilniaus Gedimino technikos universitetas
El. p.: grkdesign@gmail.com, simona.ramanauskaite@vgtu.lt

Įvadas

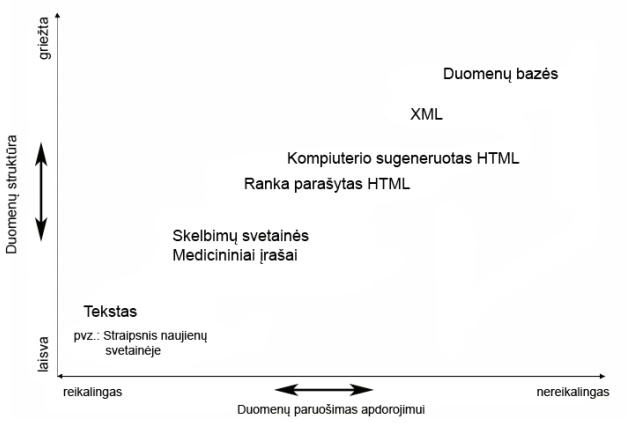
Neretai kuriant naują projektą, kuriame naudojami dažnai atsinaujinantys duomenys, vienas iš pagrindinių reikalavimų yra kuo didesnis sistemos automatizavimas, kad sistema reikalautų minimalios priežiūros. Įgyvendinant šį reikalavimą naudojami duomenų analizatoriai (angl. *parsers*) ir žiniatinklio duomenų gavyba (angl. *web mining*) [2, 3, 8, 10, 11, 12, 16].

Duomenų analizatorių paskirtis yra analizuoti jiems pateikiamų duomenų srautą, siekiant išfiltruoti reikiamus duomenis [13]. Analizuojant duomenis, duomenų nagrinėjimo algoritmai ieško tam tikrų vietų duomenyse, kurios atitiktų iš anksto nustatytus paieškos šablonus [6]. Tačiau didžioji dalis egzistuojančių sprendimų remiasi duomenų paieška pagal jų pateikimo vietą, kas apsunkina jų išgavimą, jei laikui bėgant puslapio dizainas kinta.

Šio darbo ***tikslas*** – sumažinti duomenų išrinkimo iš interneto puslapių algoritmų jautrumą, pasiūlant duomenų išrinkimo algoritmą, kuris būtų kuo mažiau priklausomas nuo interneto puslapio dizaino ar jo pokyčių ir remtųsi duomenų struktūros nustatymu, atsižvelgiant į interneto puslapyje pateikiamų ir sistemai iš anksto žinomų duomenų panašumą.

1. Duomenų išgavimas interneto puslapiuose

Informacinėse sistemose kaupiami ir pateikiami duomenys turi atitinkamą struktūrą. Aiški ir žinoma struktūra yra būtina duomenų analizės metu, o vartotojui pateikiami duomenys dažnai gali būti tik pusiau struktūruoti, nes žmogus, atsižvelgdamas į kontekstą ir turimą patirtį, geba struktūruoti nevisiškai aiškios struktūros duomenis. Analizuojant įvairius duomenų aprašymo ir pateikimo formatus, būdus, pastebima, kad tie duomenys, kurie yra lengvai apdorojami kompiuteriu, turi griežtesnę struktūrą nei tie, kurie yra sunkiau apdorojami (žr. 1 pav.).



1 pav. Įvairių e. duomenų kategorizavimas [9]

Pasauliniame žiniatinklyje pateikiami duomenys yra laikomi iš dalies struktūruotais, nes informacija dažniausiai pateikiama tekstine forma, pritaikyta žmogui, o ne kitų sistemų patogiai analizei. Todėl, norėdamas išrinkti duomenis, vartotojas turi sudaryti taisykles, nusakančias, kokių duomenų ir kur reikėtų ieškoti [18].

Šiuo metu egzistuoja keletas turinio analize paremtų algoritmų, skirtų duomenų sąrašams išgauti iš interneto puslapių, kurie gali būti skirstomi į euristinius [17], paremtus šablonais [9], ir panašumo išskyrimo [11].

D. Buttler'io, L. Liu ir C. Pu siūlomas *Omini* algoritmas [1] skirstomas į tris fazes. Toks skirstymas į fazes būdingas didžiajai daliai duomenų išrinkimo algoritmų. Pirmoji fazė – gaunamas interneto puslapio HTML kodas ir paruošiamas tolesniam apdorojimui, antroji fazė – vykdomas ieškomų duomenų vietos nustatymas, trečioji – išrenkami visi reikiami duomenys pagal nustatytas duomenų buvimo vietas. Duomenims išrinkti svarbiausia dalis dažniausiai yra antroji fazė.

Omini algoritme remiamasi euristiniais metodais (pasikartojančių, gimininigų žymių, dalinių kelių

ar jų skirtukų, standartinio pasiskirstymo atradimo), kuriais nustatomi atskirus įrašus skiriantys elementai [17]. Tačiau šis algoritmas nėra labai tikslus ir gali būti taikomas tik paprastais atvejais, kai interneto puslapis nenaudoja sudėtingos įrašų išdėstymo struktūros.

Šablonais paremti algoritmai stengiasi rasti panašias sekas, liudijančias apie kelių panašių įrašų pateikimą. Šio tipo duomenų išrinkimo algoritmai skiriasi savo lankstumu ir efektyvumu, atsižvelgiant į tai, kokiais būdais yra bandoma nustatyti panašias sekas. IEPAD [4] algoritmas taiko tapačios struktūros sekų aptikimą, o DeLa [5] naudoja lankstesnį sprendimą, kai sekos gali sutapti tik iš dalies. Tai šiam algoritmui leidžia išrinkti įrašus, net jei nėra identiškos struktūros, o IEPAD [4] algoritme bet koks pateikiamų duomenų struktūros pokytis mažina algoritmo efektyvumą.

Siekiant neprisirišti prie konkrečios struktūros ar įrašus skiriančio kodo, taikomi panašumu paremti algoritmai. MDR [13] algoritmas panašumo ieško tarp duomenų tipų, Gengxin'o Miao ir kitų siūlomas algoritmas naudoja kelio iki duomenų klasterizavimą įrašų sąrašui identifikuoti [14], ClustVX [7] ieško panašumų tarp pateikiamų įrašų įforminimo stilių įrašams identifikuoti. Šie panašumu paremti algoritmai teigia esantys tikslesni nei šablonus ar euristinius metodus naudojantys, todėl ir toliau yra plėtojami bei tobulinami.

2. Įrašo duomenų panašumu paremtas duomenų išrinkimo iš interneto puslapio algoritmas

Sistemos, kurių paskirtis yra nuolat stebėti kituose interneto puslapiuose pateikiamus duomenis, gali remtis interneto puslapyje pateikiamų duomenų panašumu į sistemai jau žinomus duomenis. Toks sprendimas leidžia identifikuoti interneto puslapyje esančius duomenų įrašus ne pagal jų vietą, o pagal reikšmę.

Šiame straipsnyje siūlomas interneto puslapiuose pateikiamų duomenų stebėjimo algoritmas, susidedantis iš 5 pagrindinių fazių:

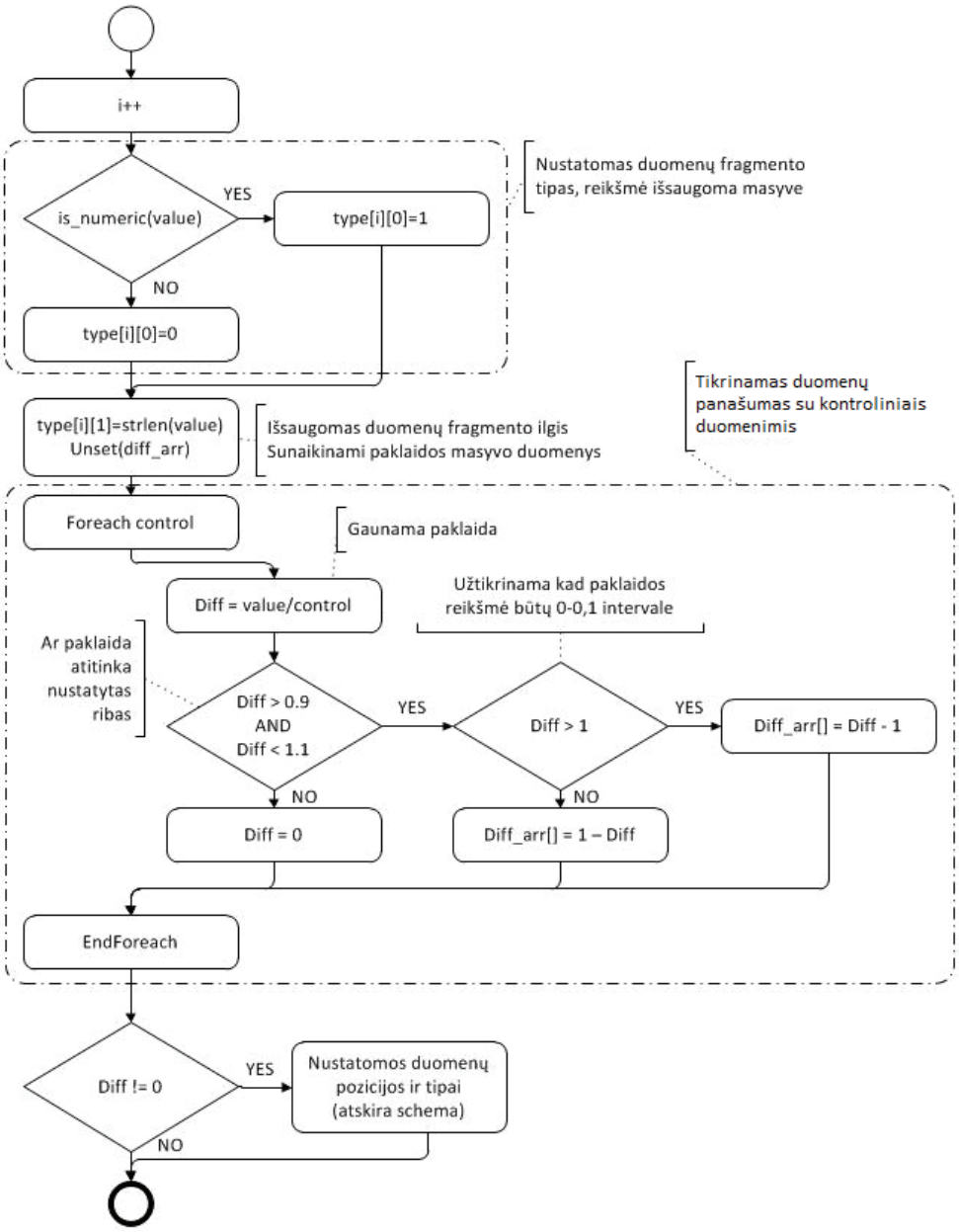
- 1 fazė: duomenų šaltinio paruošimas atlikti apdorojimą.
- 2 fazė: duomenų blokų identifikavimas, remiantis kontroliniais duomenimis.
- 3 fazė: duomenų bloko struktūros nustatymas, remiantis kontrolinių duomenų įrašo struktūra ir puslapyje pateikiamų reikšmių panašumu.
- 4 fazė: duomenų įrašų vizualaus išdėstymo nustatymas.
- 5 fazė: visų duomenų įrašų nuskaitymas, atsižvelgiant į ieškomų duomenų struktūrą ir jų išdėstymo tvarką.

Pirma ir penkta fazės sutampa su daugeliu kitų duomenų išrinkimo algoritmų, nes jų metu yra pašalinama didžioji dalis reikšmės neturinčių duomenų, duomenis išrenkant nereikšmingų duomenų blokų, elementų, o žinant, kur yra aprašomi reikiami duomenys, jie surenkami ir naudojami pagal poreikį.

Esminę siūlomo algoritmo dalį aprašo 2–4 fazės. Ji skiriasi nuo kitų šiuo metu egzistuojančių algoritmų, nes nereikalauja įrašų struktūros nurodymo, bet drauge identifikuoja įrašų struktūrą pagal duomenų panašumą.

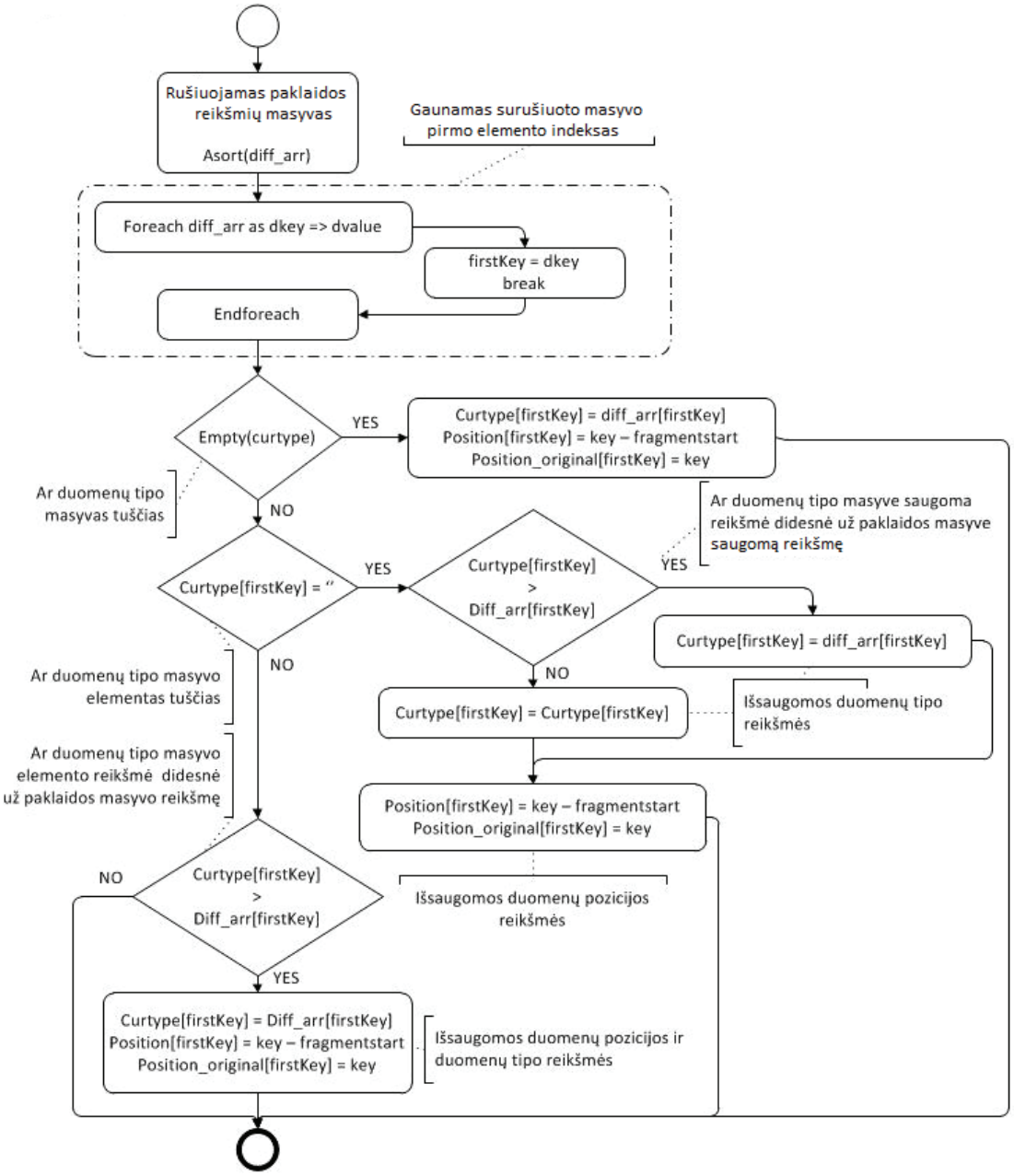
Antroje fazėje būtina turėti bent vieną kontrolinį įrašą, kuriuo remiantis identifikuojamas įrašų pateikimo blokas. Tačiau šiam kontroliniam įrašui keliamas reikalavimas, kad jis kuo mažiau kistų analizuojamuose interneto puslapiuose, pavyzdžiui, valiutų kursų stebėjimo atveju kontrolinis įrašas gali būti euro kursas, susidedantis iš valiutos kodo, jos pirkimo ir pardavimo kursų.

Analizuojant duomenims išrinkti paruoštą HTML kodą, ieškoma panašiausių duomenų į kontrolinį įrašą. Norint palyginti, galima naudoti ne visus įrašo duomenis, o tik tuos, kurie labiausiai apibūdina įrašą ar yra mažiausiai kintantys. Duomenų palyginimo algoritmas pateiktas 2 pav.



2 pav. Duomenų palyginimo su kontroliniais duomenimis algoritmas

Radus kontrolinio įrašo duomenis, interneto puslapyje yra identifikuojama visa pateikiamo įrašo struktūra, nurodant, kur yra aprašomos konkrečios duomenų įrašo savybės (žr. 3 pav.). Taip, pasinaudojant algoritmu, gaunamas įrašo aprašymo šablonas analizuojamame puslapyje.



3 pav. Duomenų pozicijų nustatymo algoritmas

Nustačius įrašų struktūros pavyzdį, ketvirtoje fazėje yra identifikuojamas įrašų išdėstymo interneto puslapyje šablonas. Duomenų įrašų išdėstymo nustatymas remiasi atstumo skaičiavimu tarp skirtingų įrašo laukų. Tam kiekvienai žemiausio lygio žymei su duomenimis suteikiamas indeksas *i*, o, identifikavus kontrolinio įrašo duomenis (jų įraše yra *d*; *d* turi būti > 1), yra skaičiuojama skirtumų tarp kontrolinio įrašo duomenų buvimo vietos žymių indeksų suma *s* (žr. 1 formulę):

$$s = \sum_{j=1}^{d} i(j) \qquad (1)$$

Jei suma *s* yra artima maksimaliam žymių indeksui ar yra didesnė nei pusė maksimalaus indekso *i* reikšmės, laikoma, kad visi įrašai yra išdėstyti horizontaliai, o jei ne – vertikaliai.

3. Duomenų išrinkimo algoritmo tyrimas

Norint nustatyti gaunamų duomenų efektyvumą, naudojami keli rodikliai [15]:

- Bendras duomenų fragmentų skaičius interneto puslapyje n_d.
- Bendras išrinktų duomenų fragmentų skaičius n_r.
- Teisingai išrinktų duomenų fragmentų skaičius (*True-Positive*) n_t.
- Neteisingai išrinktų duomenų fragmentų skaičius

(*False-Positive*) n_k. Prie šio rodiklio priskiriami ir tie duomenų fragmentai, kurie turėjo būti išrinkti algoritmo, bet dėl kažkokių priežasčių pasirinkti nebuvo (*False-Negative*).

Žinant šias duomenų išrinkimo savybes, galima apskaičiuoti, kokia yra ieškomų duomenų dalis tarp visų atrinktų duomenų *p* (žinoma kaip *precision*, žr. 2 formulę) ir kokia yra atrinktų duomenų dalis tarp visų ieškomų duomenų *r* (žinoma kaip *recall*, žr. 3 formulę). Apibendrinus galima pasakyti, jog algoritmai su aukštu *precision* rodikliu atrenka daugiau teisingų duomenų fragmentų negu klaidingų, o algoritmai su aukštu *recall* rodikliu atrenka didžiąją dalį ieškomų duomenų fragmentų:

$$p = \frac{n_t}{n_t + n_k} \tag{2}$$

$$r = \frac{n_t}{n_d} \tag{3}$$

F-score rodiklis *F* naudojamas norint apibendrinti algoritmų efektyvumo savybes ir leidžia išreikšti algoritmo efektyvumą vienu matu (žr. 4 formulę):

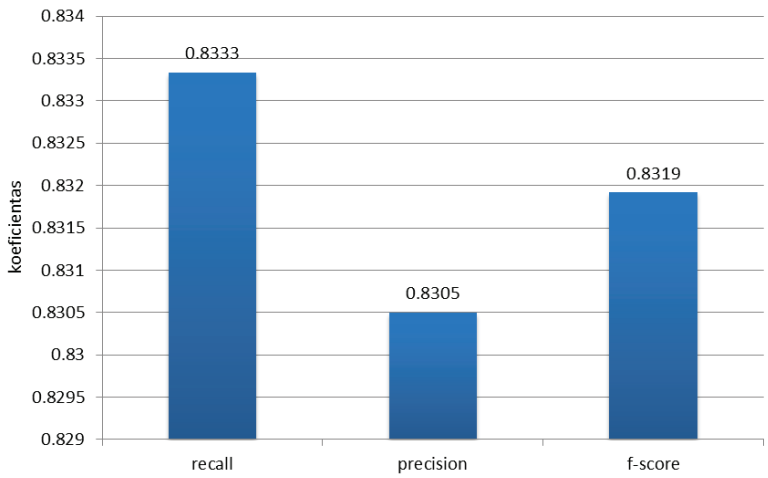
$$F = \frac{p}{r} \tag{4}$$

3.1. Siūlomo algoritmo efektyvumo tyrimas

Šio tyrimo tikslas – nustatyti siūlomo algoritmo efektyvumo rodiklius, esant skirtingam duomenų išdėstymui interneto puslapyje.

Tam buvo pasirinkti Lietuvoje ir Latvijoje veikiantys bankai ir jų teikiami duomenys apie valiutų kursus. Iš viso tyrimui naudojama 10 žiniatinklio puslapių ir dvi modifikuotos jų versijos, siekiant padidinti duomenų pateikimo variantų kiekį bei kardinaliai pakeisti duomenų išdėstymo HTML kodo struktūrą. Kaip kontrolinių duomenų rinkinys buvo pasirinktas euro kursas (pirkimo kursas – 3,4460, pardavimo kursas – 3,4590).

Testavimo metu išrinkus visų valiutų kursus iš 12 pasirinktų interneto puslapių, gauti rodikliai yra labai artimi vienas kitam. Tai reiškia, kad algoritmas veikia efektyviai, nepriklausomai nuo pateikiamų duomenų išdėstymo. Apibendrinant tyrimą ir išvedant visų tyrimų vidurkį pastebima, kad *precision*, *recall* ir *F-score* reikšmės yra lygios 0,83 (žr. 4 pav.). Tai reiškia, kad algoritmu yra atrenkama daugiau nei 80 proc. visų reikiamų įrašų, o algoritmo atrenkamų duomenų pertekliškumas nesiekia 20 proc.



4 pav. Siūlomo algoritmo efektyvumo rodikliai atrenkant valiutų kursus ir kaip kontrolinius duomenis naudojant euro kursą

3.2. Duomenų atrankos įrankių lyginamoji analizė

Siekiant palyginti, ar 0,83 *F-score* rodiklis yra tinkamas, lyginant jį su kitais algoritmais, buvo atliktas papildomas tyrimas, kurio metu siekiama ištirti pasiūlyto algoritmo efektyvumo rodiklius ir palyginti su kitų algoritmų rodikliais.

Atliekant duomenų atrankos įrankių lyginamąją analizę, analizuojami duomenų išrinkimo rezultatai, gauti atlikus duomenų atranką su trimis komerciniais įrankiais (VWR – *Visual Web Ripper*[1]; HS – *Helium Scraper*[2]; OWH – *OutWit Hub*[3]) ir sukurtu pasiūlyto algoritmo prototipu (SA). Nors šiuo metu egzistuoja nemažai įrankių ir duomenų išrinkimo iš interneto

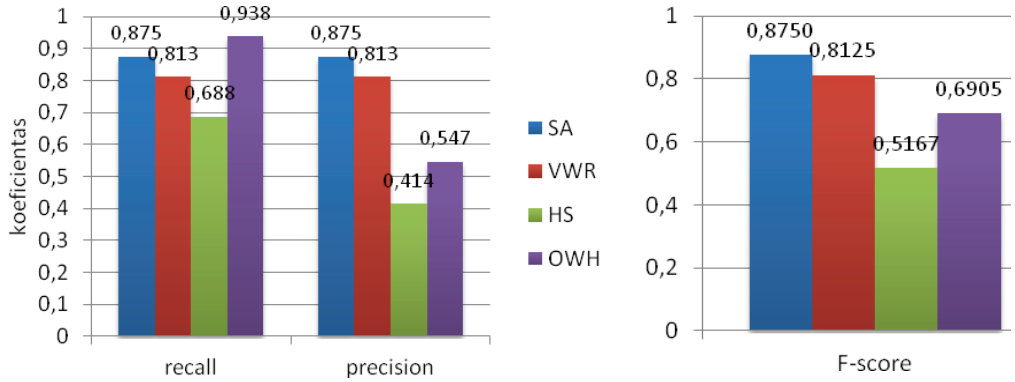
[1] Žr.: http://visualwebripper.com.

[2] Žr.: http://www.heliumscraper.com.

[3] Žr.: http://www.outwit.com.

puslapių algoritmų, tačiau ne visus įrankius ar aprašomus algoritmus įmanoma ištirti praktiškai dėl jų neatskleidžiamo kodo ar veikimo detalių.

Tyrimo metu buvo bandoma išgauti visi valiutų kursų duomenis interneto puslapyje, nurodant, koks yra kiekvienos valiutos kodas ir kokie yra jų kursai. Visa kita informacija yra ignoruojama. Tyrimas buvo vykdomas su trimis interneto puslapiais, keičiant jų dizainą ir duomenis ir taip sudarant 16 skirtingų testinių situacijų. Šio tyrimo metu gauti efektyvumo rodikliai pateikti 4 paveiksle.



5 pav. Pagal duomenų išrinkimo sistemas ir pasiūlytą algoritmą sukurto prototipo efektyvumo rodiklių palyginimas

Tyrimo rezultatai parodo, kad pagal pasiūlytą algoritmą sukurtas prototipas atrenka mažiausiai perteklinių duomenų, o teisingai atrenkamų duomenų dalimi nusileidžia tik OWH sistemai. Nors šio tyrimo metu buvo naudojama mažesnė testinių atvejų imtis ir pagal pasiūlytą algoritmą sukurto prototipo *F-score* rodiklis gautas didesnis nei ankstesnio tyrimo metu, bet galima teigti, kad suminis pasiūlyto algoritmo efektyvumo rodiklis yra geriausias, lyginant su kitais įrankiais, ir kad nė vienas iš kitų įrankių nepasiekė 0,83 *F-score* rodiklio.

Išvados

1. Analizė parodė, kad naujausi duomenų išrinkimo algoritmai labiau remiasi panašumų kode nustatymu, o ne konkrečių šablonų naudojimu. Tai leidžia sukurti lankstesnes duomenų išrinkimo sistemas, ne taip prisirišančias prie interneto puslapio dizaino ir duomenų pateikimo formos.
2. Analizuoti panašumo išskyrimo duomenų išrinkimo algoritmai siekia nustatyti duomenų pateikimo įrašus pagal jų aprašymui reikalingo kodo panašumą. Mūsų siūlomas algoritmas remiasi duomenų įrašų panašumo nustatymu. Tai leidžia padidinti algoritmo nepriklausomumą nuo interneto puslapio duomenų pateikimo formos, tačiau apriboja algoritmo naudojimą, skiriant jį stebėti duomenis arba atlikti iš dalies žinomų duomenų paiešką.
3. Sukurtas algoritmas pasižymi dideliu efektyvumu (vidutinė *F-score* reikšmė lygi 0,83). Šio algoritmo prototipas rodo vienodus *recall* ir *precision* rodiklius (0,875), įrodydamas savo stabilumą perteklinių ir ieškomų duomenų atrankos atžvilgiu, o *precision* reikšme lenkia visus kitus analizuotus įrankius.

Literatūra

1. Buttler D., Liu L., Pu C., 2001, A fully automated object extraction system for the World Wide Web. *Distributed Computing Systems*. 21st International Conference on IEEE.
2. Castellano M., et al., 2007, A web text mining flexible architecture. *World Academy of Science, Engineering and Technology*. 32. P. 78–85.
3. Chang Chia-Hui, et al., 2006, *A survey of web information extraction systems.* IEEE Transactions on Knowledge and Data Engineering. 18 (10). P. 1411–1428.
4. Chang Chia-Hui, Shao-Chen Lui, 2001, *IEPAD: information extraction based on pattern discovery*. Proceedings of the 10th international conference on World Wide Web. ACM.
5. Devika K., Surendran S., 2013, An overview of web data extraction techniques. *International Journal of Scientific Engineering and Technology.* 2 (4).
6. di Buono M. P., 2015, *Semi-automatic Indexing and Parsing Information on the Web with NooJ*. International NooJ Conference. Springer International Publishing.
7. Grigalis T., 2012, Towards Automatic Structured Web Data Extraction System. *DB&Local Proceedings*.
8. Youssefi A. H., Duke D. J., Zaki M. J., 2004, *Visual web mining*. Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters. ACM.
9. Yuanyuan Liao, Wang Jianhu, 2011, Research on text

mining. *American Journal of Engineering and Technology Research*. Vol. 11 (9).
10. Kosala R., Blockeel H., 2000, Web mining research: A survey. *ACM Sigkdd Explorations Newsletter*. 2 (1). P. 1–15.
11. Kushmerick N., 2003, Finite-state approaches to web information extraction. *Information Extraction in the Web Era*. Springer, Berlin, Heidelberg. P. 77–91.
12. Laender A. H. F., et al., 2002, A brief survey of web data extraction tools. *ACM Sigmod Record*. 31 (2). P. 84–93.
13. Liu Bing, Grossman R., Yanhong Zhai, 2003, *Mining data records in web pages*. Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM.
14. Miao G., et al., 2009, *Extracting data records from the web using tag path clustering*. Proceedings of the 18th international conference on World wide web. ACM.
15. Powers D. M. W., 2015, *Evaluation Evaluation a Monte Carlo study.* arXiv preprint arXiv:1504.00854.
16. Zhao Qiankun, Bhowmick S. S., 2003, *Sequential pattern mining: A survey*. Technical Report CAIS Nayang Technological University Singapore. P. 1–26.
17. Zhai Y., Liu B., 2005, *Web data extraction based on partial tree alignment*. Proceedings of the 14th international conference on World Wide Web. ACM.
18. Wu, Yu-Chieh, 2016, Multilingual news extraction via stopword language model scoring. *Journal of Intelligent Information Systems*. P. 1–23.

Summary

AN ALGORITHM FOR DATA EXTRACTION FROM WEB PAGES BASED ON DATA SIMILARITIES

K. Griazev, S. Ramanauskaitė

Problems with data extraction from web pages were analysed, a proposed solution is provided in the paper. Analysis showed that data-based algorithms are more popular than path-based data extraction. We propose a new data retrieval algorithm based on web page data similarity to controlled data.

The efficiency of the proposed data retrieval algorithm was applied to the retrieval of currency exchange rates data, the efficiency of this algorithm prototype was evaluated by comparing it to other products. Research showed that the proposed data retrieval algorithm, although more suitable for the retrieval of constantly changing data and requires controlled data, is more efficient than other similar products.

Keywords: data extraction, data parsing, data similarity.

Santrauka

DUOMENŲ IŠRINKIMO INTERNETO PUSLAPIUOSE ALGORITMAS, PAREMTAS DUOMENŲ TARPUSAVIO PANAŠUMU

K. Griazev, S. Ramanauskaitė

Straipsnyje aprašoma ir analizuojama duomenų išrinkimo iš interneto puslapių problema ir egzistuojantys sprendimai. Atsižvelgiant į šiuo metu vyraujančias tendencijas duomenims išrinkti naudoti ne keliu, o duomenimis paremtus algoritmus, pasiūlytas naujas duomenų išrinkimo iš interneto puslapių algoritmas, paremtas kontrolinių duomenų panašumu į interneto puslapyje pateikiamus duomenis ir taip leidžiantis vienu kontroliniu įrašu nustatyti visų duomenų įrašų padėtį puslapyje.

Straipsnyje analizuojamas pasiūlyto algoritmo efektyvumas, jį taikant valiutų kursų duomenų stebėjimo atveju, ir įvertintas šio algoritmo sukurto prototipo efektyvumas, lyginant jį su kitais šiuo metu rinkoje siūlomais įrankiais. Šių tyrimų rezultatai parodo, kad siūlomas duomenų išrinkimas, nors ir yra skirtas labiau stebėti nuolat kintančius duomenis bei reikalauja kontrolinių įrašų, vis dėlto pasižymi didesniu efektyvumu nei kiti analogiški produktai.

Prasminiai žodžiai: duomenų išrinkimas, duomenų atranka, duomenų panašumas.

Įteikta 2017-03-19
Priimta 2017-06-23