Lietuvių kalbos homografų vienareikšminimas remiantis leksemų ir morfologinių pažymų vartosenos dažniais

«Disambiguation of Lithuanian Homographs Based on the Frequencies of Lexemes and Morphological Tags»

by Pijus Kasparaitis; Tomas Anbinderis

Source:
Studies About Languages (Kalbų Studijos), issue: 14 / 2009, pages: 25-31, on www.ceeol.com.

KOMPIUTERINĖ LINGVISTIKA/ COMPUTATIONAL LINGUISTICS

Lietuvių kalbos homografų vienareikšminimas remiantis leksemų ir morfologinių pažymų vartosenos dažniais

Tomas Anbinderis, Pijus Kasparaitis

Anotacija. Norint sintezuoti balsą iš teksto, tekstą reikia sukirčiuoti. Problema ta, kad egzistuojantys lietuvių kalbos automatinio kirčiavimo algoritmai kai kuriems žodžiams (homografams) pateikia daugiau negu vieną kirčiavimo variantą. Šiame darbe homografams vienareikšminti pritaikytas iki šiol lietuvių kalbai nenaudotas metodas, pagrįstas leksemų ir morfologinių pažymų vartosenos dažniais, gautais iš vieno milijono žodžių tekstyno. Tekstynas iš pradžių buvo sukirčiuotas automatiškai, po to pakoreguotas rankiniu būdu. Homografai vienareikšminami atmetant rečiau vartojamas gramatines formas ir leksemas. Papildomų sunkumų sukelia tas faktas, kad vienas žodis gali atitikti daugiau negu dvi gramatines formas. Šios problemos sprendimui pasiūlyta skaičiuoti gramatinių formų porų dažnius. Darbe parodyta, kad morfologinių pažymų dažniai yra svarbesni už leksemų dažnius. Pasiūlyti metodai leido homografus vienareikšminti 85,01% tikslumu. Nors šie metodai nesinaudoja jokia informacija apie žodžio kontekstą, pasiekti rezultatai panašūs į kontekstą naudojančio algoritmo ID3 rezultatus.

***Reikšminiai žodžiai**: teksto kirčiavimas; tomografai; vienareikšminimas; leksema; morfologinė pažyma; balso sintezė.*

Įvadas

Automatinis teksto kirčiavimas yra vienas iš balso sintezės pagal tekstą etapų. Lietuvių kalbos žodžių kirčiavimo algoritmai jau nagrinėti (Kasparaitis, 2000, 2001; Kazlauskienė ir kt., 2004; Norkevičius ir kt., 2004). Žodžio kirčiavimą patogu išskaidyti į tokius tris žingsnius: antraštinio pavidalo (lemos) suradimas, gramatinio aprašo (giminė, skaičius, linksnis ir pan.) suradimas ir kirčio vietos bei priegaidės nustatymas remiantis lema bei gramatiniu aprašu. Nemaža dalis lietuvių kalbos žodžių gali turėti kelias lemas ir kelis gramatinius aprašus. Pavyzdžiui, žodis *galvos* gali būti: a) daiktavardžio *galva* vienaskaitos kilmininkas; b) daugiskaitos vardininkas; c) veiksmažodžio *galvoti* būsimasis laikas III asmuo. Tokie žodžiai vadinami homoformomis, o pats reiškinys – morfologiniu daugiareikšmiškumu (Rimkutė, 2002). Darbe (Rimkutė, 2002) teigiama, kad homoformos sudaro 39% lietuviško teksto, o darbe (Rimkutė ir Grybinaitė, 2004) – kad net 47%. Dėl morfologinio daugiareikšmiškumo kai kurių žodžių negalima vienareikšmiškai sukirčiuoti, todėl reikalingas vienareikšminimas (vieno varianto išrinkimas) Kai kurie homoformų vienareikšminimo (ar daugiareikšmiškumo ribojimo) algoritmai jau nagrinėti E. Rimkutės ir bendraautorių darbuose (Rimkutė ir Grybinaitė, 2004, 2006a, 2006b): statistiniai slaptieji Markovo modeliai, loginis ID3 algoritmas, sintaksinė analizė.

Dalis homoformų tariamos vienodai, dalis – skirtingai. Skirtingai tariamos homoformos vadinamos homografais. Darbe (Kasparaitis, 2000) nagrinėtuose tekstuose homografai sudarė daugiau negu 15% visų žodžių, darbe (Rimkutė, 2002) – 8,1% (812 žodžių iš 10000). Šiame darbe bus nagrinėjamos ne visos homoformos, o tik homografai ir jų vienareikšminimas, nes tik homografai sukelia sunkumų kirčiavimo algoritmą pritaikant balso sintezei pagal tekstą. Vienareikšminimui bus naudojami duomenys apie leksemų ir morfologinių aprašų vartosenos dažnius, tokie problemos sprendimo būdai lietuvių kalbai dar nebuvo naudoti (Rimkutė ir Grigonytė, 2006b). Taigi šio **darbo tikslas** – pasiūlyti naują paprastą algoritmą homografams vienareikšminti, kuris tikslumu nenusileistų kitiems metodams.

1. Duomenų atranka ir paruošimas

Dideliam kiekiui tekstų sukirčiuoti buvo panaudotas automatinio kirčiavimo algoritmas, kuris išsamiai aprašytas darbe (Kasparaitis, 2000). Šio algoritmo čia neaprašinėsim, paminėsim tik tai, kad algoritmas naudoja tris žodynus: daiktavardžių-būdvardžių (toliau DB), veiksmažodžių (toliau Vks), nekaitomų žodžių (toliau Nek). Šių žodynų įrašus vadinsime leksemomis, kiekviename įraše saugomas žodžio kamienas, linksniuotė/ asmenuotė, kirčiuotė ir kita kaitymui ir kirčiavimui reikalinga informacija. Iš kiekvienos leksemos galime nesunkiai rasti žodžio antraštinį pavidalą (lemą), todėl toliau straipsnyje kartais vietoje leksemos vartosime tik žodžio lemą, o kartais – tik kamieną, turėdami galvoje, kad pašalinus kamieną pašalinama ir visa leksema.

Minėtas algoritmas buvo įkomponuotas į specialią programą, kuri sukirčiuoja tekstą, skirtingomis spalvomis išskiria kelis kirčiavimo variantus turinčius ir nekirčiuotus žodžius, leidžia vartotojui parinkti vieną kirčiavimo variantą ar pataisyti (uždėti) kirčio ženklą. Su šia programa

profesionalus filologas sukirčiavo ir peržiūrėjo aibę tekstų, kurių bendra apimtis beveik milijonas žodžių (985967 žodžiai). Tekstai buvo surinkti iš interneto ir pagal žanrą suskirstyti į šešias grupes: grožinė literatūra, mokslinė literatūra, įstatymai, respublikinė periodika, vietinė periodika, specializuota ir populiarioji periodika. Parenkant tekstus pagal žanrą, buvo stengiamasi atsižvelgti į VDU tekstyne (prieiga per internetą http://donelaitis.vdu.lt, žiūrėta 2008-10-23) esančias proporcijas. Laikantis tokių pačių proporcijų, tekstai buvo padalinti į penkias maždaug vienodas dalis. Keturios dalys buvo naudojamos mokymui, o viena dalis – testavimui.

Duomenys tolesniems eksperimentams gaunami taip: imama po vieną kirčiuoto teksto žodį, šis žodis (pašalinus kirčio ženklą) pateikiamas kirčiavimo algoritmui. Kirčiavimo algoritmas sugeneruoja visas galimas hipotezes, kokio žodžio kokia gramatinė forma tai gali būti ir kaip ji kirčiuojama. Toliau šiame straipsnyje trumpumo dėlei vadinsime tiesiog **hipotezėmis**. Kadangi minėti kirčiuoti tekstai buvo sudaryti ne specialiai šiems eksperimentams, o kitais tikslais, buvo išsaugota tik informacija apie kirčiavimą, informacijos apie lemas ir gramatines formas nėra. Lygindami kirčiavimo algoritmo generuotų hipotezių kirčiavimą su kirčiuotu tekstu, randame hipotezes, kurių kirčiavimas sutampa su kirčiuotu tekstu (vad. **teisingomis hipotezėmis**), ir kurių nesutampa (vad. **klaidingomis hipotezėmis**). Pavyzdžiui, kirčiuotame tekste sutikus žodį *galvõs*, teisingos hipotezės bus a ir c (žr. pirmą įvado pastraipą), o b – klaidinga hipotezė.

Kadangi vienas žodis gali atitikti dvi, tris ir dar daugiau gramatinių formų (hipotezių), be to, nuspėti gramatinių formų skaičių ir galimus jų derinius yra sunku, todėl nagrinėsime tik gramatinių formų (hipotezių) poras, kur viena hipotezė yra teisinga, o kita klaidinga. Pavyzdžiui, žodžiui *galvos* tokių hipotezių porų būtų dvi (a-b ir b-c). Kiekvienai hipotezių porai naudodami kirčiuotą tekstą galime suskaičiuoti, kiek kartų teisinga buvo pirmoji hipotezė, ir kiek kartų – antroji. Hipotezę, kuri daugiau kartų buvo teisinga, vadinsime **dažnesne hipoteze**, o kitą poros hipotezę – **retesne hipoteze**. Rašydami hipotezių poras dažnesnę hipotezę rašysime pirmiau. Vienareikšminant homografus bus tiesiog imamos hipotezių poros ir atmetamos retesnės hipotezės, taip tikintis, kad liks tik tos, kurios kirčiuojamos vienodai.

Dabar išsamiau pasižiūrėkime, kokią informaciją saugo kiekviena hipotezė: 1) žodynas (DB, Vks, Nek); 2) leksema (leksemos identifikatorius); 3) gramatinė forma. Nekaitomų žodžių gramatinę formą charakterizuoja leksemos identifikatorius, todėl šie parametrai sutampa. Daiktavardžių-būdvardžių gramatinę formą charakterizuoja du parametrai: linksniuotė ir skaičius/ linksnis. Taigi galime laikyti, kad kirčiavimo algoritmas užpildo hipotezių porų lentelę. Pavyzdžiui, jei visas turimas kirčiuotas tekstas sudarytas tik iš tokių dviejų žodžių *Mamà galvõs*, tuomet kirčiavimo algoritmo darbo rezultatas atrodys kaip pavaizduota 1 lentelėje. Stulpeliuose Ar_teis1 ir Ar_teis2 loginės reikšmės rodo atitinkamai teisingą ir klaidingą hipotezę. Veiksmažodžiams ir nekaitomiems žodžiams stulpelis Gr_f_12 arba Gr_f_22 yra tuščias.

Tolesni eksperimentai bus atliekami įvairiais pjūviais grupuojant duomenis, analogiškus pateiktiems 1 lentelėje.

2. Leksemų atmetimas

Iš pradžių panagrinėkime, kaip vienareikšminimui galima panaudoti leksemų dažnius. Galima pastebėti, kad kai kurių leksemų buvimas žodyne labiau kliudo kirčiuoti, nei padeda. Pavyzdžiui, darbe (Rimkutė ir Grybinaitė, 2004) paminėtas itin retai vartojamas žodis *kokis* (žodžio *kokybė* sinonimas), kurio vienaskaitos kilmininkas sutampa su gana dažnai vartojamo įvardžio *koks* vienaskaitos kilmininku, o šie žodžiai kirčiuojami skirtingai (skiriasi priegaidė). Išmetus tokią retai vartojamą leksemą iš žodyno, daugiau žodžių būtų galima sukirčiuoti vienareikšmiškai.

Bus nagrinėjami du leksemų atmetimo būdai:

1) randamos daiktavardžių-būdvardžių žodyno leksemų poros, kurių sutampa kamienai ir linksniuotės. Tai galima užrašyti tokia užklausa:

   SELECT Leksema1, count(Ar_teis1 = TRUE),
   Leksema2, count(Ar_teis2 = TRUE),
   GROUP BY Leksema1, Leksema2,
   WHERE (Žodynas1 = „DB“) && (Žodynas2 = „DB“)
   && (Gr_f_11 = Gr_f_21) && (Gr_f_12 = Gr_f_22).

   Šioje užklausoje reikalavimas, kad sutaptų ir skaičiai/ linksniai, ir linksniuotės, garantuoja, kad sutampa kamienų tekstinis pavidalas. Analogiškai, kaip apibrėžėme dažnesnę hipotezę, galime apibrėžti **dažnesnę leksemą**. Dažnesne vadinsime tą, iš kurios dažniau generuojamos teisingos hipotezės. Atmetamos retesnės leksemos. Toliau pateiktuose pavyzdžiuose vietoj leksemos rašysime lemą, šalia lemos pateiksime pasikartojimų skaičių, trumpumo dėlei atskirsime dažnesnę leksemą nuo retesnės ženkliuku >. Pavyzdžiui, *kláusimas* (699) > *klausìmas* (0), *Jõnas* (172) > *jònas* (17), *gė́rimas* (72) > *gėrìmas* (2), *romãnas* (49) > *Ròmanas* (5). Analogiškai galima atmesti ir nekaitomus žodžius, kurie retesni už kitus nekaitomus žodžius, pvz., *paskuĩ* (156) > *pãskui* (11), arba už kaitomų žodžių gramatines formas, pvz., *mètro* (12) > *metrò* (9), *dõmino* (7) > *dominò* (0).

1 lentelė. Hipotezių porų lentelė

Žodynas1	Gr_f_11	Gr_f_12	Leksema1	Ar_teis1	Žodynas2	Gr_f_21	Gr_f_22	Leksema2	Ar_teis2
DB	vns. V.	3 linksn.	mama	TRUE	DB	vns. Š.	3 linksn.	mama	FALSE
DB	vns. Įn.	3 linksn.	mama	TRUE	DB	vns. Š.	3 linksn.	mama	FALSE
DB	vns. K.	3 linksn.	galva	TRUE	DB	dgs. V.	3 linksn.	galva	FALSE
Vks	būs. l.	–	galvoti	TRUE	DB	dgs. V.	3 linksn.	galva	FALSE

2) kiekvienai leksemai tiesiog suskaičiuojamas teisingų (klaidingų) hipotezių skaičius. Tai nusakoma užklausa:

SELECT Leksema1, count(Ar_teis1 = TRUE), Leksema2, count(Ar_teis2 = TRUE), GROUP BY Leksema1, Leksema2.

Analogiškai atmetamos retesnės leksemos.

Kalbant apie čia pateiktas užklausas, reikia nepamiršti, kad hipotezių tvarka porose yra atsitiktinė, todėl poras sugrupavus pagal leksemas, kiekvienai grupei galima rasti simetrišką grupę, t. y. tokią, kurios dešinę pusę sukeitę su kaire gausime identišką grupę. Tokias grupes reikia sujungti sukeitus kairę pusę su dešine.

Pirmasis būdas leidžia atmesti tik nedidelį skaičių leksemų, tačiau jis garantuoja, kad nepadaugės nekirčiuotų ar klaidingai kirčiuotų žodžių. Antrasis būdas leidžia atmesti daugiau leksemų, tačiau jis gali turėti ir pašalinių efektų. Pavyzdžiui, iš veiksmažodžio kamieno *kárti* (*kãria*, *kórė*) padarytas dalyvis *kártas* sutampa su daiktavardžiu *kařtas*, tačiau *kařtas* (1911) > *kártas* (104). Analogiškai veiksmažodžio *laisvė́ti* (*laisvė́ja*, *laisvė́jo*) būsimasis laikas *laisvẽs* sutampa su daiktavardžio *láisvė* vienaskaitos kilmininku ar daugiskaitos vardininku, tačiau *láisvės* (140) > *laisvẽs* (1). Atmetus šiuos veiksmažodžių kamienus, neliks problemų kirčiuojant daiktavardžius *kartas* ir *laisvė*, tačiau liks nekirčiuoti kiti iš šių veiksmažodžių kamienų daromi žodžiai. 2 lentelėje pateikti duomenys, kiek kuris būdas leido atmesti leksemų ir kiek tos leksemos mokymo duomenims generavo teisingų (klaidingų) hipotezių ir teisingų hipotezių dalis procentais.

Kalbant apie nekaitomus žodžius dar verta paminėti, kad klitikai (žodžiai, kurių nereikia kirčiuoti) pradiniame kirčiavimo algoritmo variante ir pritaikius pirmą būdą buvo atpažįstami naudojant specialų algoritmą (Anbinderis ir Kasparaitis, 2007), o naudojant antrą būdą jie tiesiog buvo pašalinti iš nekaitomų žodžių sąrašo.

Kaip matome iš 2 lentelės, atmetus kai kurias leksemas teisingų hipotezių dalis tarp visų hipotezių padidėja 6,1%, tačiau, kokią įtaką šis padidėjimas turi teksto kirčiavimui, bus tiriama ketvirtame skyriuje.

3. Gramatinių formų dažniais grįstos taisyklės

3.1. Taisyklių formavimas

Šiame skyriuje panagrinėsime 1 lentelės duomenų grupavimą pagal gramatines formas; tam naudojama tokio tipo užklausa:

SELECT Žodynas1, Gr_f_11, Gr_f_12, count(Ar_teis1 = TRUE), Žodynas2, Gr_f_21, Gr_f_22, count(Ar_teis2 = TRUE), GROUP BY Žodynas1, Gr_f_11, Gr_f_12, Žodynas2, Gr_f_21, Gr_f_22.

Gaunam naują hipotezių porų (dažnesnės ir retesnės) lentelę, kurios įrašus galima traktuoti kaip tam tikras vieno kirčiavimo varianto parinkimo remiantis morfologinių požymų dažniais taisykles, turinčias tokį pavidalą:

(Žodynas1, Gr_f_11, Gr_f_12) > (Žodynas2, Gr_f_21, Gr_f_22).

Suskirstykime gautas taisykles į keturias grupes prie aukščiau minėtos užklausos pridedant papildomų sąlygų:

A. Užklausa dar papildyta sąlyga

WHERE ((Žodynas1 = „Nek“) || (Žodynas2 = „Nek“),

kuri į atskirą grupę išskiria visas su nekaitomais žodžiais susijusias taisykles.

B. Iš likusių taisyklių naudojant papildomą sąlygą

WHERE ((Žodynas1 = Žodynas2) && (Leksema1 = Leksema2))

išskirtos taisyklės, apimančios to paties kaitomo žodžio skirtingas gramatines formas.

C. Išskirtos taisyklės, gautos iš skirtingų kaitomų žodžių žodynų. Papildoma sąlyga:

WHERE (Žodynas1 <> Žodynas2).

D. Visos likusios taisyklės. Šiuo atveju žodynai sutampa, kaip ir B grupėje, tačiau skiriasi leksemos.

3.2. Taisyklių grupių analizė

Dabar išsamiau panagrinėkime taisyklių grupes, jų turinį dar kartą sugrupavę pagal žodynų poras (žr. 3 lentelę).

2 lentelė. Leksemų atmetimo rezultatai mokymo duomenims

		Pradinis	**Pritaikius 1 būdą**		**Pritaikius 1 ir 2 būdus**	
			Liko	**Atmesta**	**Liko**	**Atmesta**
DB	**Leksemų skaičius**	62106	62041	65	61107	999
	Hipotezių skaičius	594344/ 102505 85,3%	593549/ 98277 85,8%	795/ 4228 15,8%	586654/ 63160 90,3%	7690/ 39345 16,3%
Vks	**Leksemų skaičius**	8826	8826	0	8437	389
	Hipotezių skaičius	243340/ 82934 74,6%	243340/ 82934 74,6%	0/ 0	235346/ 46919 83,4%	7994/ 36015 18,2%
Nek	**Leksemų skaičius**	2049	2038	11	1915	134
	Hipotezių skaičius	61211/ 1274 98,0%	61119/ 919 98,5%	92/ 355 20,6%	61081/ 715 98,8%	130/ 559 18,9%
	Iš viso	898895/ 186713 82,8%	898008/ 182130 83,1%	887/ 4583 16,2%	883081/ 110794 88,9%	15814/ 75919 17,2%

3 lentelė. Taisyklių grupių turinys

Taisyklių grupė	Žodynų pora	Taisyklių skaičius	Teisingų hipotezių skaičius	Klaidingų hipotezių skaičius	Teisingų hipotezių procentas	Žodžių skaičius vienai taisyklei
A	DB>Nek	5	185	83	69,0%	53,6
	Nek>DB	44	2712	74	97,3%	63,3
	Vks>Nek	3	16	0	100%	5,3
	Nek>Vks	31	7885	41	99,5%	255,7
	Nek-Nek	3	315	22	93,5%	84,3
	Iš viso	**87=>2**	**11113**	**220**	**98,1%**	130,3
B	DB-DB	104	52690	12702	80,6%	628,8
	Vks-Vks	64	33911	6914	83,1%	637,9
	Iš viso	**168**	**86601**	**19616**	**81.5%**	632.2
C	DB>Vks	293	41007	4752	89,6%	156,2
	Vks>DB	209	16726	4468	78,9%	101,4
	Iš viso	**502**	**57733**	**9220**	**86.2%**	133.4
D	DB-DB	388	37074	7391	83,4%	114,6
	Vks-Vks	155	16040	2366	87,1%	118,7
	Iš viso	**543**	**53114**	**9757**	**84,5%**	115,8
Iš viso		**1215**	**208561**	**38813**	**84.3%**	

Iš 3 lentelės A grupės matome, kad labai paprastai galima išspręsti nekaitomų žodžių vienareikšminimo problemą: jei koks nors nekaitomas žodis retesnis už kitą nekaitomą žodį ar kaitomo žodžio formą, tokį žodį galima tiesiog išmesti iš žodyno (1, 3, 5 eilutės). Šį veiksmą galima perkelti į anksčiau minėtą leksemų atmetimo etapą. Likusiems atvejams kirčiuoti gauname dvi itin paprastas taisykles: (Nek, *, *) > (DB, *, *) ir (Nek, *, *) > (Vks, *, *), kurios rodo, kad nekaitomas žodis dažnesnis už bet kokią gramatinę formą, padarytą iš daiktavardžio-būdvardžio ar veiksmažodžio kamieno. Toks sprendimas leidžia teisingai parinkti kirčiavimo variantą net 98,1% tikslumu.

Mes norime, kad taisyklės būtų kuo labiau apibendrinančios, t. y. sudarytos iš kuo didesnio skaičiaus pavyzdžių. Kiek pavyzdžių vidutiniškai apibendrina viena taisyklė, nurodyta dešiniajame 3 lentelės stulpelyje. Kaip matome B grupės taisyklės yra labiau apibendrinančios (vidutiniškai 632,2 pavyzdžiai), negu kitų grupių. Tai rodo, kad homografai dažniausiai yra to paties žodžio gramatinės formos, ko ir buvo galima tikėtis. Tačiau B grupę sudaro nedaug taisyklių ir jų tikslumas mažiausias.

Kalbant apie C grupę, galima pastebėti, kad daiktavardžio-būdvardžio gramatinės formos dažnesnės už veiksmažodžių gramatines formas, todėl 502 (293+209) taisykles pakeitus viena taisykle (DB, *, *) > (Vks, *, *) vis vien tikslumas būtų 67,9% (41007 + 4468)/ (41007 + 4468 + 16726 + 4752).

Sudarant D grupės taisykles, iš veiksmažodžių žodyno buvo gautos ir 163 taisyklės, kuriose abi gramatinės formos sutapo. Žinoma, tokios taisyklės neturi jokios prasmės, todėl jos nebuvo įtrauktos į 3 lentelę.

Taigi sumažinus A grupės taisyklių skaičių iki 2, gauname 1215 taisyklių rinkinį, kuris mokymo duomenims leidžia teisingą kirčiavimo variantą parinkti 84,3% tikslumu.

3.3. Rezultatų palyginimas

Įdomu palyginti sukurtų taisyklių tikslumą su rezultatais, gautais naudojant kitus algoritmus. Palyginimui tinkamų rezultatų pavyko rasti tik darbe (Rimkutė ir Grybinaitė, 2004), tiesa, jame eksperimentai atlikti su kitais duomenimis ir buvo vienareikšminamos visos, o ne tik skirtingai kirčiuojamos, gramatinės formos. Vienareikšminimui buvo panaudotas ID3 algoritmas, kuris remiasi gretimų žodžių morfologiniais požymiais. Šiame darbe sukurtos taisyklės buvo testuojamos naudojant 200000 žodžių tekstus, kurie nebuvo naudoti sukuriant taisykles. Rezultatų palyginimas pateiktas 4 lentelėje. Kiekvienai linksniuotei (linksniavimo paradigmai) buvo sukurta po vieną taisyklę. Taisyklių skaičius pateiktas lentelės dešiniajame stulpelyje, pvz., pirmoje eilutėje nurodyta, kad septyniolikai linksniavimo paradigmų dažnesnis buvo vienaskaitos kilmininkas, o dviem linksniavimo paradigmoms – daugiskaitos vardininkas.

Kaip matyti iš 4 lentelės, rezultatai gana panašūs, gal tik nežymiai blogesni.

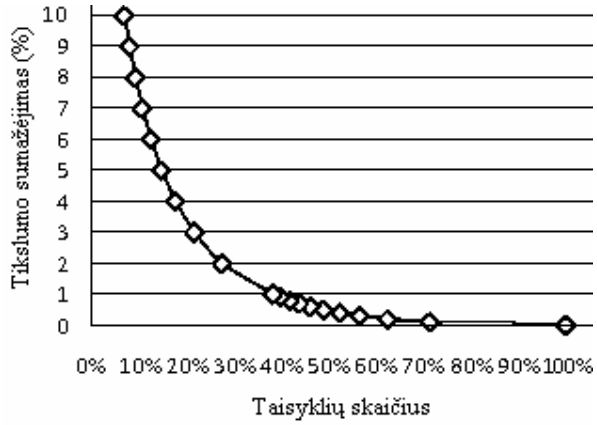
4 lentelė. ID3 algoritmo ir dažniais grįstų taisyklių palyginimas

Sutampančių gramatinių formų pora	**Tikslumas taikant ID3 algoritmą**	**Tikslumas taikant dažniais grįstas taisykles**	**Taisyklių skaičius**
Moteriškosios giminės vienaskaitos kilmininkas ir daugiskaitos vardininkas	73,65%	77,47%	17+2
Moteriškosios giminės vienaskaitos vardininkas ir įnagininkas	81,65%	74,73%	4+4
Bendratis ir neveikiamosios rūšies būtojo laiko neįvardžiuotinių vyriškosios giminės dalyvių daugiskaitos vardininkas	92,15%	91,98%	1

3.4. Taisyklių skaičiaus sumažinimas

Skirtumas tarp B ir D grupės taisyklių yra tas, kad B grupėje leksemų identifikatoriai sutampa, o D grupėje skiriasi. Palyginus šias grupes rasta, kad 53 taisyklės sutampa, o septyniais atvejais B ir D grupėse buvo viena kitai priešingos taisyklės (t. y. tokios, kurios sutaptų sukeitus kairę pusę su dešine). Sutampančias taisykles galima sujungti. Vietoj dviejų priešingų taisyklių palikus tik vieną, dažniau vartojamą, tikslumas sumažėtų, tačiau nežymiai (mažiau nei 0,1%). Taigi galima taisyklių skaičių sumažinti 60 taisyklių arba 4,9%.

Kadangi likusių taisyklių skaičius vis vien gana didelis (1155) ir skiriasi taisyklių naudojimo dažnis, tai įdomu panagrinėti, kaip keičiasi kirčiavimo tikslumas mažinant taisyklių skaičių, t. y. atsisakant taisyklių, kurioms skirtumas tarp teisingų ir klaidingų hipotezių skaičiaus yra mažiausias. Atsisakius kai kurių taisyklių, daliai žodžių nebus surasta nė vienos taisyklės. Laikysim, kad tokiems žodžiams vieną kirčiavimo variantą parinksim atsitiktinai ir taip parinkdami suklysim 50% atvejų (nes daugumą homografų galima kirčiuoti dviem būdais). Kai naudojamos visos taisyklės, tikslumas yra 84,3%. 1 pav. pavaizduota, kiek taisyklių galima atsisakyti, kad tikslumas sumažėtų tam tikru fiksuotu procentu (nuo 0,1 iki 10%).



1 pav. Tikslumo priklausomybė nuo taisyklių skaičiaus

Iš 1 pav. matome, kad palikus 40% taisyklių, tikslumas sumažės tik 0,9%, o palikus tik 7% taisyklių – sumažės 10%.

4. Teksto kirčiavimo eksperimentai

Kuo didesnis tam tikros taisyklės teisingų hipotezių kiekis (procentais) mokymo duomenyse, tuo tikslesnio testinio teksto kirčiavimo galime tikėtis. Priklausomybė tarp teisingų hipotezių procento ir teksto kirčiavimo tikslumo nėra tiesioginė. Tai lemia ne tik neatitikimas tarp mokymo ir testinių duomenų, bet ir tas faktas, kad kai kuriems žodžiams generuojama daugiau negu dvi (teisinga ir klaidinga) hipotezės, bet ir sudėtingesni jų deriniai, todėl ir jų sąveika tampa sudėtingesnė. Kaip taisyklės sąveikauja ir kokį realiai kirčiavimo tikslumą duoda, galima sužinoti tik jas naudojant tekstui kirčiuoti. Buvo atlikti trys kirčiavimo eksperimentai su trim skirtingais taisyklių, kurios remiasi hipotezių dažniais, rinkiniais: 1) jokios taisyklės nenaudojamos, 2) grupės A, B ir C, 3) grupės A, B, C ir D. Kiekvienas iš šių eksperimentų pakartotas po du kartus: pirmuoju atveju žodžiai, turintys daug kirčiavimo variantų, laikomi klaidingai kirčiuotais, antruoju atveju imamas pirmasis kirčiavimo variantas (tai ekvivalentu atsitiktiniam vieno varianto parinkimui). Visi minėti kirčiavimo eksperimentai pakartoti po tris kartus su skirtingais žodynais: 1) naudoti pilni žodynai, 2) naudotas pirmasis leksemų atmetimo būdas, 3) naudotas pirmasis ir antrasis būdai. Hipotezių dažniais pagrįstos taisyklės visuose eksperimentuose buvo sudaromos tik iš pilnų žodynų (t. y. netaikant jokio leksemų atmetimo algoritmo), nors leksemų atmetimo poveikis jau pajuntamas kirčiavimo algoritmui grąžinant kirčiavimo variantus. Taigi iš viso atlikta 18 eksperimentų. Gauti rezultatai užima daug vietos ir nėra labai vaizdūs. Mus labiausiai domina, kiek žodžių leidžia vienareikšminti vienas ar kitas algoritmas ir kokiu tikslumu jis tai daro. Tuo tikslu iš teksto išsirinksime tik žodžius, kuriuos galima kirčiuoti keliais būdais ir bent vienas kirčiavimo variantas teisingas, suskaičiuosime, kiek žodžių iš šio sąrašo galima vienareikšminti naudojant tam tikrą algoritmą ir kokia dalis buvo vienareikšminta teisingai.

Pirmiausiai panagrinėkime, kaip veikia hipotezių dažniais pagrįstos taisyklės lyginant su atveju, kai taisyklės nenaudotos. Jos taikomos žodžiams, likusiems pritaikius leksemų atmetimo algoritmą. Atsitiktinis varianto parinkimas nenaudotas. Rezultatai 5 lentelėje.

Iš 5 lentelės matome, kad: 1) taikant leksemų atmetimo 1 algoritmą tikslumas padidėja 0,2–0,3%, o taikant 1 ir 2 – sumažėja 0,8–1,1%; 2) grupės A, B ir C paveikia žymiai daugiau žodžių nei D; 3) taikant grupę D tikslumas nežymiai krenta (iki 0,3%).

Dabar panagrinėkime, kokį tikslumą galima pasiekti, jei po visų algoritmų taikymo iš likusių variantų vieną parinksime atsitiktinai. Remdamiesi intuicija, manytume, kad tikslumas turėtų būti apie 50%. Rezultatai 6 lentelėje. Juos galima būtų apibendrinti taip: kuo mažiau variantų likę (išskyrus atvejį, kai naudojamos tik grupės A, B ir C), tuo didesnį tikslumą duoda atsitiktinis parinkimas. Tai rodo, kad tarp likusių hipotezių daugiau teisingų hipotezių.

5 lentelė. Vienareišminimas naudojant hipotezių dažniais pagrįstas taisykles

Hipotezių dažniais paremtos taisyklės	**Leksemų atmetimo algoritmas nenaudotas**		**Leksemų atmetimo 1 algoritmas**		**Leksemų atmetimo 1 ir 2 algoritmai**	
	Vienareikšminta žodžių	**Teisingų**	**Vienareikšminta žodžių**	**Teisingų**	**Vienareikšminta žodžių**	**Teisingų**
Grupės A, B ir C	24179	84,9%	24155	85,2%	19934	84,1%
Grupės A, B, C ir D	29170	84,9%	29138	85,1%	20810	83,8%

6 lentelė. Vienareišminimas atsitiktinai parenkant vieną kirčiavimo variantą

Hipotezių dažniais paremtos taisyklės	Leksemų atmetimo algoritmas nenaudotas		Leksemų atmetimo 1 algoritmas		Leksemų atmetimo 1 ir 2 algoritmai	
	Vienareikšminta žodžių	Teisingų	Vienareikšminta žodžių	Teisingų	Vienareikšminta žodžių	Teisingų
Taisyklės nenaudotos	30197	52,5%	29517	52,3%	20967	63,5%
Grupės A, B ir C	6018	44,7%	5362	45,4%	1033	52,7%
Grupės A, B, C ir D	1027	53,3%	379	67,8%	157	68,2%

7 lentelė. Vienareišminimas naudojant leksemų atmetimo algoritmus

Hipotezių dažniais paremtos taisyklės	Leksemų atmetimo 1 algoritmas		Leksemų atmetimo 1 ir 2 algoritmai			
	Vienareikšminta žodžių	Teisingų	Vienareikšminta žodžių	Teisingų	Padaugėjo nekirčiuotų	Apibendrintas tikslumas
Taisyklės nenaudotos	681	91,04%	8928	95,71%	1266	84,24%
Grupės A, B ir C	740	92,03%	5569	92,48%	1213	76,56%
Grupės A, B, C ir D	739	92,15%	1986	85,10%	1204	54,29%

Galiausiai buvo įvertinta leksemų atmetimo algoritmų įtaka. Tačiau 2 algoritmas gali paveikti ne tik kelis kirčiavimo variantus turinčius žodžius, tačiau ir kitus žodžius, pavyzdžiui, kirčiuotą žodį paversti nekirčiuotu. Žinoma, nedidelę dalį žodžių iš tikrųjų reikia palikti nekirčiuotą. Taigi 2 algoritmui greta vienareikšminimo tikslumo dar buvo skaičiuojamas apibendrintas tikslumas, kurį galima užrašyti formule: $(tv+tn)/(tv+tn+kv+kn)$, kur

tv – teisingai vienareikšmintų žodžių skaičius,
tn – teisingai paliktų nekirčiuotais žodžių skaičius,
kv – klaidingai vienareikšmintų žodžių skaičius,
kn – klaidingai paliktų nekirčiuotais žodžių skaičius.

Rezultatai 7 lentelėje.

Galima daryti išvadą, kad leksemų atmetimo 2 algoritmą verta taikyti tik tuo atveju, jei nenaudojamos taisyklės.

Turint tikslą visiems daug kirčiavimo variantų turintiems žodžiams parinkti vieną variantą geriausi rezultatai pasiekti, kai iš pradžių taikytas leksemų atmetimo 1 algoritmas, tada taisyklių grupės A, B, C ir D ir galiausiai iš likusių atsitiktinai parinktas vienas variantas. Kiek žodžių vienareikšmino kiekvienas algoritmas ir kokiu tikslumu, pateikta 8 lentelėje.

8 lentelė. Geriausius rezultatus davusi vienareikšminimo algoritmų seka

Algoritmas	Vienareikšminta žodžių	Teisingų
Leksemų atmetimo 1 algoritmas	680	91,18%
Taisyklių grupės A, B, C ir D	29138	85,09%
Atsitiktinis varianto parinkimas	379	67,81%
Iš viso	30197	85,01%

Taigi iš viso testiniuose tekstuose buvo 30197 homografai (15,30% visų žodžių), teisingą variantą pavyko nustatyti 85,01% tikslumu.

Išvados

Šiame darbe pasiūlyta homografų vienareikšminimui pasinaudoti leksemų ir morfologinių pažymų dažniais, iš likusių variantų vieną parinkti atsitiktinai. Pasiekti rezultatai leidžia daryti tokias išvadas:

- Atmetus kai kurias leksemas, mokymo duomenims teisingų hipotezių dalis tarp visų hipotezių padidėja 6,1%.
- Sudarytas morfologinių pažymų dažniais grįstų taisyklių rinkinys (1215 taisyklių), kuris mokymo duomenims leidžia teisingą kirčiavimo variantą parinkti 84,4% tikslumu.
- Jei naudojamos morfologinių pažymų dažniais grįstos taisyklės, verta atmesti tik tas leksemas, kurių sutampa ir kamienai, ir linksniuotės. Atmetus daugiau leksemų, rezultatai pablogėja (daugiau žodžių lieka nekirčiuota).
- Iš galimų kirčiavimo hipotezių atmetus rečiau vartojamas, tarp likusių hipotezių didėja teisingą kirčiavimą nusakančių hipotezių dalis (net iki 68%), todėl iš likusių hipotezių vieną variantą verta parinkti atsitiktinai.
- Nors morfologinių pažymų dažniais grįstose taisyklėse nenaudojama jokia informacija apie kontekstą, tačiau jos geba vienareikšminti kai kurias gramatines formas tikslumu, artimu kontekstinę informaciją naudojančiam ID3 algoritmui.
- Pritaikius pasiūlytus algoritmus teksto kirčiavimui, homografus pavyko vienareikšminti 85,01% tikslumu.

Padėka

Autoriai dėkoja filologei Rūtai Bagužytei už tai, kad sukirčiavo milijono žodžių tekstyną, ir UAB „Žinių amžius“ už finansinę paramą šio tekstyno rengimui.

Literatūra

1. Anbinderis, T., Kasparaitis, P., 2007. Klitikų paieškos lietuviškame tekste algoritmai. *Kalbų studijos*, nr. 10, pp.30–37.
2. Kasparaitis, P., 2000. Automatic Stressing of the Lithuanian Text on the Basis of a Dictionary. *Informatika*, no. 11 (1), pp.19–40.
3. Kasparaitis, P., 2001. Automatic Stressing of the Lithuanian Nouns and Adjectives on the Basis of a Rules. *Informatika*, no. 12 (2), pp.315–336.
4. Kazlauskienė, A., Norkevičius, G., Raškinis, G., 2004. Automatizuotas lietuvių kalbos veiksmažodžių kirčiavimas: problemos ir jų sprendimas. *Baltų ir kitų kalbų fonetikos ir akcentologijos problemos*, pp.166–173.
5. Norkevičius, G., Kazlauskienė, A., Raškinis, G., 2004. Bendrinės lietuvių kalbos daiktavardžių ir būdvardžių kirčiavimo struktūrinis modelis, algoritmas ir realizacija. *Kalbų studijos*, nr. 6, pp.72–76.
6. Rimkutė E., 2002. Homoformos dabartinės lietuvių kalbos tekstyne. *Lituanistika*, nr. 2 (50), pp.86–101.
7. Rimkutė, E., Grybinaitė, A., 2004. Dažniausios lietuvių kalbos morfologinio daugiareikšmiškumo rūšys ir jų automatinis vienareikšminimas. *Kalbų studijos*, nr. 5, pp.74–78.
8. Rimkutė, E., Grigonytė, G., 2006a. Automatizuotas lietuvių kalbos morfologinio daugiareikšmiškumo ribojimas. *Kalbų studijos*, nr. 9, pp.30–37.
9. Rimkutė, E., Grigonytė, G., 2006b. Statistiniai, loginiai ir kompiuterių mokymosi metodai lietuvių kalbos morfologiniam daugiareikšmiškumui riboti – konferencijos *Informacinės technologijos 2006*, pranešimų medžiaga. Kaunas: Technologija, pp.104–108.

Tomas Anbinderis, Pijus Kasparaitis

Disambiguation of Lithuanian Homographs Based on the Frequencies of Lexemes and Morphological Tags

Summary

In the text-to-speech synthesis it is necessary to stress the text. The main problem is that currently existing algorithms of stress for Lithuanian produce more than a single stressing possibility for some words (homographs). The method based on frequency of occurrences of certain lexemes and morphological tags was proposed in this work. Such method has never been used for Lithuanian. The frequencies were calculated using text corpus containing 1 million words. Text corpus was stressed automatically and then corrected manually. Disambiguation of homographs is performed by removing less frequently used grammatical forms and lexemes. Additional problems arise due to the fact that a single word can correspond to more than two grammatical forms. The method based on the frequencies of pairs of grammatical forms was proposed in this work. It was shown that the frequencies of morphological tags play more important role than the frequencies of lexemes. The method proposed allows disambiguating the homographs with the accuracy of 85.01%. Despite the fact that the method proposed does not employ contextual information, the results achieved are comparable with those achieved with the algorithm ID3 that uses the context.

Straipsnis įteiktas 2008 11
Parengtas spaudai 2009 04

Apie autorius:

Tomas Anbinderis, Vilniaus universiteto informatikos doktorantas, Vilniaus universiteto Matematikos ir informatikos fakulteto Kompiuterijos katedros asistentas.
Mokslinės veiklos sritys: balso sintezė iš teksto, kitos kompiuterinės lingvistikos sritys.
Adresas: Vilniaus universitetas, Matematikos ir informatikos fakultetas, Kompiuterijos katedra, Naugarduko g. 24, 03225 Vilnius.
El. paštas: Tomas.Anbinderis@mif.vu.lt

Pijus Kasparaitis, dr. (fiziniai mokslai), Vilniaus universiteto Matematikos ir informatikos fakulteto Kompiuterijos katedros lektorius. 2001 m. apgynė daktaro disertaciją „Lietuvių kalbos kompiuterinė sintezė“.
Mokslinės veiklos sritys: balso sintezė iš teksto, kitos kompiuterinės lingvistikos sritys.
Adresas: Vilniaus universitetas, Matematikos ir informatikos fakultetas, Kompiuterijos katedra, Naugarduko g. 24, 03225 Vilnius.
El. paštas: pkasparaitis@yahoo.com