Postupci morfološke normalizacije u pretraživanju informacija i klasifikaciji teksta

Jan Šnajder
Zavod za elektroniku, mikroelektroniku, računalne i inteligentne sustave
Fakultet elektrotehnike i računarstva Sveučilišta u Zagrebu
Unska 3, HR-10000 Zagreb, Hrvatska
E-mail: jan.snajder@fer.hr

***Sažetak*—Zbog morfologije jezika riječi se u tekstu pojavljuju u različitim morfološkim oblicima. Ta pojava negativno utječe na performanse sustava za pretraživanje informacija i klasifikaciju teksta, osobito kod morfološki složenih jezika. Istraživanja su pokazala da je performanse moguće poboljšati provođenjem nekog oblika morfološke normalizacije, odnosno sažimanjem različitih morfoloških varijanti riječi na jedan jedinstveni oblik. Postupci morfološke normalizacije već su desetljećima predmetom istraživanja u području pretraživanja informacija, a istraživanja su danas dodatno intenzivirana sve većom potrebom za učinkovitim višejezičnim pretraživanjem informacija. U ovome radu dan je pregled najpoznatijih, ali i nekih novijih postupaka morfološke normalizacije, te je opisan uobičajeni način njihova vrednovanja.**

***Ključne riječi*—Morfološka normalizacija, lematizacija, korjenovanje, pretraživanje informacija, klasifikacija teksta, obrada prirodnog jezika.**

I. Uvod

Većina metoda pretraživanja informacija (engl. information retrieval, IR) i klasifikacije teksta (engl. text classification, TC) temelji se na *modelu vektorskoga prostora* [1] u kojemu su tekstovni dokumenti prikazani zbirkom pozicijski neovisnih riječi, odnosno tzv. *vrećom riječi* (engl. bag-of-words). Za takve metode učinci morfologije jezika mogu biti vrlo nepovoljni: umjesto da je predstavljen jednim jedinim oblikom, koncept se raspršuje na više različitih morfoloških varijanti. Budući da se pojam iz upita (engl. query term) u dokumentima može pojavljivati u različitim morfološkim varijantama, pri pretraživanju informacija može se dogoditi da relevantni dokumenti budu nisko rangirani, ili čak da uopće ne budu dohvaćeni. Pri klasifikaciji teksta, morfološke varijacije smanjuju pouzdanost statističkih procjena i stoga degradiraju ukupnu performansu sustava. Pored toga, morfološka varijacija povećava dimenzionalnost prostora značajki, koja je za većinu postupaka klasifikacije teksta već ionako dovoljno problematična [2].

Morfološku varijaciju, odnosno strukturu riječi sazdanu od morfema, proučava lingvistička disciplina *morfologije*. Morfologiju je moguće podijeliti u dvije široke grane: *flektivnu* i *derivacijsku*. Prva se bavi fleksijom (engl. inflection), odnosno tvorbom oblika riječi (engl. word-forms). Oblici riječi izražavaju gramatičke značajke riječi, primjerice broj i padež kod imenica; broj, padež, rod i stupanj kod pridjeva, i sl. Fleksija se uobičajno ostvaruje dodavanjem *afikasa* (gramatičkih morfema) osnovi riječi (dijelu riječi koji nosi leksičko značenje). Većinom su to *sufiksi* odnosno nastavci (afiksi koji se nadovezuju na osnovu riječi), a rjeđe *prefiksi* (afiksi koji prethode osnovi). Primjer fleksije u hrvatskome jeziku jest *kuća→kućom* ili *bogat→najbogatijih*. Predmet proučavanja derivacijske morphologije jest tvorba novih riječi iz postojećih uporabom derivacijskih afikasa. Primjer derivacije u hrvatskom jeziku je *kuća→kućanski* ili *bogat→bogatstvo*. Premda derivacija mijenja značenje riječi, uobičajeno su izvedenica i polazna riječ barem donekle semantički srodne.

Kako bi se uklonili negativni utjecaji morfološke varijacije na pretraživanje informacija i klasifikaciju teksta, uobičajeno se u takvim sustavima primjenjuje neki oblik *morfološke normalizacije*. Morfološka normalizacija podrazumijeva sažimanje različitih morfoloških varijanti jedne riječi, bilo flektivnih ili derivacijskih, na jedan jedinstveni oblik, morfološku *normu*. U osnovi, taj postupak možemo smatrati specijalnim slučajem postupka *sažimanja pojmova* (engl. term conflation), odnosno *grupiranja pojmova* (engl. term clustering), koji je međutim temeljen samo na morfološkoj, a ne i na leksičko-semantičkoj varijaciji [3]. U području pretraživanja informacija, morfološka normalizacija često se smatra tehnikom za povećanje odziva (engl. recall), ali i preciznosti pri niskim razinama odziva budući da relevantnim dokumentima može omogućiti probijanje do višeg ranga [4].

U ovom radu dan je pregled postupaka morfološke normalizacije i načina njihovog vrednovanja. Ostatak rada organiziran je na sljedeći način. U II. dijelu diskutiraju se ukratko utjecaji morfološke varijacije na sustave za pretraživanje informacija i klasifikaciju teksta. U III. dijelu dana je podjela pristupa morfološkoj normalizaciji te su u glavnim crtama opisani neki tipični pristupi. Vrednovanje postupaka normalizacije opisano je u IV. dijelu, a u V. dijelu iznesen je zaključak.

II. Utjecaj morfološke varijacije

Utjecaj morfologije na performanse sustava za pretraživanje informacija i klasifikaciju teksta to je veći što je veća morfološka složenost jezika. Morfološka složenost jezika određena je učestalošću afiksacije u jeziku te složenošću postupka segmentacije afikasa u pojedinim oblicima riječi [5]. Neki jezici (npr. vijetnamski) uopće ne koriste afiksaciju, dok je u drugima (npr. japanskom) afikse lako segmentirati. Kod jezika u kojima je učestalost afiksacije veća, veća je i morfološka

varijacija, pa je i potreba za morfološkom normalizacijom u načelu izraženija. Učinkovitost samog postupka morfološke normalizacije određena je pak složenošću segmentacije afikasa.

Zbog toga što gramatičke značajke uglavnom izražava česticama i poretkom riječi u rečenici, engleski je jezik, u usporedbi s drugim jezicima indo-europske skupine, razmjerno jednostavne morfološke složenosti. S druge strane, slavenski jezici, uključivo hrvatski, znatno su morfološki složeniji, posebice u pogledu fleksije. Primjerice, većina pridjeva u hrvatskome jeziku može poprimiti i više od 40 različitih oblika. Pored toga, postoji problem velikog broja višeznačnih flektivnih sufikasa, tj. sufikasa koji, ovisno o tome s kojim ih se osnovama kombinira, označavaju različite morfosintaktičke značajke. Primjerice, flektivni sufiks '-e' izražava značajke genitiva jednine većine imenica ženskoga roda (npr. *vode*, *ruke*), međutim isti sufiks kod imenica muškog roda može izražavati značajku akuzativa množine (npr. *vojnike*, *zakone*). Zbog ove višeznačnosti, slavenski su jezici bogati istopisnicama (homografima), tj. oblicima riječi koji su flektivno povezani s dva ili više različitih leksema. Primjerice, *vode* može biti oblik imenice *voda*, imenice *vod* te glagola *voditi*. Pored toga, u slavenskim su jezicima učestale glasovne promjene na granicama morfema. Primjerice, dok sufiks '-e' u obliku *vojnike* označava akuzativ množine, u *vojniče* isti je sufiks korišten u kombinaciji s glasovno izmijenjenom osnovom kako bi označio vokativ jednine. U kontekstu pretraživanja informacija i klasifikacije teksta, ovako visok stupanj morfološke složenosti vrlo je problematičan.

Premda rana istraživanja za engleski jezik nisu bila indikativna [6], danas se ipak smatra da morfološka normalizacija poboljšava pretraživanje informacija [7]. Kod morfološki složenih jezika poboljšanja uslijed primjene morfološke normalizacije u načelu su izraženija. Pokazano je da morfološka normalizacija pospješuje pretraživanje informacija u slučaju europskih jezika iz romanske (francuski, talijanski, portugalski), germanske (njemački, nizozemski, švedski) ili ugrofinske (mađarski, finski) jezične porodice [8], [9]. Za slavenske jezike, to je utvrđeno za slovenski [10] i češki [11]. Preliminarna istraživanja provedena za hrvatski jezik [12], [13] upućuju na slične zaključke.

Kao i kod pretraživanja informacija, čini se da su utjecaji morfološke normalizacije na klasifikaciju teksta temeljenu na modelu *vreće riječi* također jezično ovisni. Dok za engleski [14], njemački [14], [15] i nizozemski [16] nisu utvrđena značajna poboljšanja, kod morfološki složenijeg hrvatskog jezika poboljšanja su nešto izraženija [17]–[19]. Istraživanja također sugeriraju da je utjecaj morfološke normalizacije izraženiji pri manjem broju značajki [17], [20]. Morfološka normalizacija primjenjuje se često i u svrhu smanjenja dimenzionalnosti prostora značajki [21]. U [17] je pokazano da je kod hrvatskog jezika uporabom morfološke normalizacije u kombinaciji s odgovarajućom metodom izbora značajki dimenzionalnost prostora moguće smanjiti za čak 55%, i to bez ikakvog utjecaja na performansu sustava. Uzmemo li u obzir da prostor značajki u ovakvim slučajevima redovito poprima razmjere reda veličine 10^5, postaje jasan praktični značaj koji morfološka normalizacija ima kod postupaka za klasifikaciju teksta.

III. Pristupi morfološkoj normalizaciji

U literaturi su predložene različite taksonomije pristupa morfološkoj normalizaciji [3], [14], [22]–[25]. Sasvim općenito, postupke normalizacije možemo podijeliti prema:

- vrsti obuhvaćene morfološke varijacije na *flektivne* i *derivacijske*,
- vrsti uporabljenog znanja na *lingvističke* i *statističke*,
- načinu izgradnje na *ručne* i *automatske*,
- načinu same normalizacije na *rječničke* (eng. dictionary-based) i one temeljene na *pravilima* (eng. rule-based).

Postupci koji se oslanjaju na lingvističko znanje u načelu su *jezično ovisni*, dok su postupci temeljeni na rječniku u načelu *ograničenog opsega* jer mogu normalizirati samo one oblike koji su sadržani u rječniku.

Dva osnovna pristupa morfološkoj normalizaciji jesu *korjenovanje* (engl. stemming) i *lematizacija* (engl. lemmatisation). U užem smislu, korjenovanje podrazumijeva uklanjanje afikasa iz oblika riječi kako bi se dobio korijen zajednički svim oblicima [26]. Tako dobiven korijen ne mora odgovarati pravom korijenu riječi u lingvističkome smislu.[1] U širem smislu, korjenovanje je svaki onaj postupak koji rezultira klasama ekvivalencije morfološki povezanih riječi, tzv. *korijenskim klasama* (engl. stem classes) ili *sažimajućim skupovima* (engl. conflation sets). Korjenovanje može obuhvatiti i flektivnu i derivacijsku morfološku varijaciju te može rezultirati različitim stupnjevima normalizacije, od slabog ili *konzervativnog* do jakog ili *agresivnog* korjenovanja.

Za razliku od korjenovanja, lematizacija je postupak kojim se pronalazi lingvistički ispravan kanonski (natuknički) oblik neke riječi, odnosno *lema*. S lingvističkog stajališta, lematizacija je sofisticiranija od korjenovanja i za morfološki složene jezike srodna je problemu *morfološke analize*. Pored toga, lematizacija može uključivati razrješavanje homografije te morfosintaktičko označavanje, što je čini šire primjenjivom. U kontekstu morfološke normalizacije, dvije su osnovne razlike između lematizacije i korjenovanja. Prvo, za razliku od korjenovanja, lematizacija uvijek rezultira lingvistički ispravnom normom. Drugo, lematizacija obuhvaća samo flektivnu, dok korjenovanje može obuhvatiti i derivacijsku morfološku varijaciju. Lematizacija u načelu iziskuje više lingvističkog znanja od korjenovanja, i to se znanje izravno ili neizravno temelji na nekom obliku morfološkog leksikona.

U nastavku su detaljnije razmotreni neki tipični pristupi morfološkoj normalizaciji.

A. Rječnička lematizacija

Rječnička lematizacija (engl. dictionary-based lemmatisation) podrazumijeva uporabu ručno sastavljenih morfoloških

[1]Neki izvori koriste pojam *pseudokorijen* kako bi naglasili ovu razliku. Također valja primijetiti da je pseudokorijen u praksi sličniji morfološkoj osnovi riječi (engl. stem) nego njenom korijenu (engl. root), odnosno korijenskom (leksičkom) morfemu.

leksikona koji oblike neke riječi, eventualno označene dodatnom morfosintaktičkom informacijom, povezuju s pripadnom lemom (odnosno, u slučaju homografije, s više mogućih lema). Rječnička normalizacija teoretski nudi apsolutnu lingvističku točnost, no izgradnja morfološkog leksikona iziskuje veliko lingvističko znanje i ogroman ljudski napor. Pored toga, rječnička normalizacija nužno je ograničenog opsega i jezično ovisna. Primjer rječničke lematizacije jest lematizacijski poslužitelj [12] izgrađen nad *Hrvatskim morfološkim leksikonom* [27].[2] Posljednja verzija ovog leksikona sadržava više od 100.000 lema i gotovo 4 milijuna različitih oblika.

Problem ograničenosti opsega morfološkog leksikona i ljudskog napora potrebnog za njegovu izgradnju u velikoj mjeri rješavaju postupci (polu-)automatske akvizicije leksikona iz korpusa. Takvi postupci koriste lingvističko znanje u obliku morfoloških pravila i statističke informacije iz korpusa kako bi iz tog istog korpusa izlučili leme i povezali ih s odgovarajućim morfološkim paradigmama. Ljudski napor potreban je u tom slučaju za formalnu definiciju morfologije jezika te za eventualne povremene intervencije uzrokovane višeznačnošću jezika. Budući da je postupak akvizicije leksikona moguće ponavljati nad različitim korpusima, problem ograničenosti opsega manje je izražen. Opisani pristup korišten je za akviziciju morfološkog lekikona za francuski [28], slovački [29] i ruski jezik [30] te za proširivanje postojećeg Hrvatskog morfološkog leksikona [27]. Sličan pristup predložen je i u okviru formalizma *funkcijske morfologije* [31]. Pristup akviziciji morfološkog leksikona osmišljen posebno u svrhu morfološke normalizacije, te primijenjen na hrvatski jezik, opisan je u [32].

Problematičnom se na prvi pogled kod rječničkih postupaka može činiti brzina izvođenja. Međutim, taj se problem u praksi vrlo učinkovito rješava primjenom konačnih automata, odnosno *preobličivača* (engl. transducers) [33]. Preobličivači omogućavaju da ulazni oblik riječi preoblikuje u normalizirani oblik u vremenu proporcionalnom s brojem slova ulaznog oblika, a pored toga pružaju značajnu uštedu memorijskog prostora.

B. Korjenovanje temeljeno na pravilima

Najrašireniji pristup korjenovanju predstavljaju pristupi temeljeni na pravilima (engl. rule-based stemming), također poznati kao *algoritmi uklanjanja afikasa* (engl. affix removal algorithms). Ovi algoritmi pronalaze korijen riječi primjenom niza ručno kodiranih pravila odsijecanja (odnosno zamjene) sufikasa i prefikasa. Uobičajeno je riječ o heurističkim pravilima temeljenima na ograničenom lingvističkom znanju. Premda s računarskog stajališta algoritmi korjenovanja predstavljaju brzo i elegantno rješenje, njihovo oblikovanje može biti poprilično izazovno. To je osobito slučaj kod morfološki složenih jezika koji intenzivno koriste afiksaciju i koji obiluju glasovnim promjenama, pa stoga iziskuju veliki broj pravila.

Prvi algoritmi odsijecanja sufikasa razvijeni su za engleski jezik: *Lovinsov algoritam* [26], *Porterov algoritam* [34] i *Paice-Huskov algoritam* [35]. Lovinsov algoritam uklanja najdulji sufiks koji se podudara s jednim od sufikasa iz pripremljenog popisa. Kod Porterovog algoritma pravila odsijecanja sufiksa kaskadno su posložena u pet razina. Prvom razinom ostvaruje se flektivna, a ostalim četirima derivacijska normalizacija. Primjerice, pravilo

$$(m > 1) \quad \text{ence} \rightarrow \varepsilon$$

opisuje odsijecanje sufiksa *-ence* (npr. *inference→infer*), uz uvjet da je heuristička mjera korijena m veća od 1. Mjera m otprilike odgovara broju slogova u korijenu. Definirana je kao $[C](VC)^m[V]$, gdje V označava niz samoglasnika, C niz suglasnika, $[\cdot]$ opcionalno pojavljivanje, a m broj ponavljanja niza VC. Uvjet $m > 1$ u gornjem pravilu osigurava da ono nije primjenjivo na, primjerice, oblik *defence* (mjera m korijena *def* je 1). Na taj se način mogu spriječiti mnoge pogreške *prekorjenovanja* (engl. overstemming errors), odnosno pogreške pretjeranog sažimanja morfološki ili semantički potpuno nepovezanih oblika riječi (npr. korjenovanje oblika *defence* i *define* na korijen *def*). Paice-Huskov algoritam koristi slična pravila, no primjenjuje ih iterativno sve dok se ne dosegne zaustavno pravilo ili dok niti jedno pravilo više nije omogućeno.

Po uzoru na algoritme za engleski jezik, slični su algoritmi razvijeni za druge, morfološki složenije jezike, uljučivo amharski [36], arapski [37], nizozemski [38], [39], francuski [40], grčki [41], mađarski i portugalski [8], latinski [42], malezijski [43], ruski i ukrajinski [44], slovenski [10], poljski [45], španjolski [46] i švedski [47]. U okviru projekta *Snowball* razvijeni su i algoritmi korjenovanja za neke druge jezike.[3] Razmjerno jednostavan algoritam korjenovanja za hrvatski jezik predložen je u [48].

C. Hibridno korjenovanje

Unatoč tome što postavljaju uvjete na primjenu pojedinih pravila, kod algoritama odsijecanja ipak su česte pogreške prekorjenovanja. Na primjer, Porterov algoritam [34] pogrešno normalizira oblike *policy* i *police* na zajednički korijen *polic*. Ovaj problem donekle se može riješiti *hibridnim postupcima* koji uz pravila odsijecanja sufikasa dodatno koriste rječnik. Primjer takvog postupka je *Krovetzov algoritam korjenovanja* KSTEM [49] koji pravilo odsijecanja primjenjuje samo na one oblike koji nisu sadržani u rječniku. Dodatno, kako bi se spriječila normalizacija semantički nepovezanih oblika, KSTEM ograničava primjenu pravila odsijecanja derivacijskih sufikasa temeljem podudaranja u rječničkim definicijama (engl. gloss overlap). Ovaj pristup, inače prvi puta upotrijebljen za razrješavanje leksičke višeznačnosti [50], temelji se na pretpostavci da će semantički srodne riječi biti u rječniku definirane uporabom istih riječi. Eksperimenti nad engleskim jezikom pokazali su da algoritam KSTEM poboljšava preciznost sustava za pretraživanje informacija više nego Porterov algoritam, no da su razlike ipak male (manje od 5% promjene preciznosti).

Drugi pristup problemu prekorjenovanja semantički nepovezanih riječi predložen je u [51]. Algoritam provodi naknadno

[2] `http://hml.ffzg.hr`

[3] `http://snowball.tartarus.org`

particioniranje klasa ekvivalencije dobivenih Porterovim algoritmom na temelju statističke informacije o supojavljivanju riječi u korpusu. Particioniranje se temelji na pretpostavci da će se semantički srodni oblici blisko supojavljivati u tekstu te da samo takve oblike treba normalizirati istom normom. Primjerice, u tekstu u domeni kulture uputno je na istu normu normalizirati oblike *izdati* i *izdavač*, no to ne vrijedi univerzalno te bi u nekoj drugoj domeni moglo biti štetno. Mjera supojavljivanja riječi a i b definirana je na sljedeći način:

$$em(a,b) = \max\Big(\frac{n_{ab} - En(a,b)}{n_a + n_b}, 0\Big),$$

gdje su n_a i n_b broj pojavljivanja riječi a odnosno b u korpusu, n_{ab} je broj njihova supojavljivanja u prozoru neke određene veličine, a $En(a,b)$ je očekivani broj supojavljivanja. Particioniranje klasa ekvivalencije provodi se tako da se one najprije predoče grafom, a zatim se uklanjaju svi oni bridovi za koje je vrijednost mjere supojavljivanja ispod nekog predefiniranog praga. Ovakvim statističkim pristupom dobiva se konzervativnija te korpusu i domeni prilagođena normalizacija. Eksperimenti nad engleskim pokazali su međutim razmjerno mala poboljšanja u preciznosti sustava za pretraživanje informacija (manje od 5% promjene preciznosti).

D. Strojno učena lematizacija

Učestali pristup morfološkoj normalizaciji, odnosno lematizaciji, predstavlja uporaba metoda *nadziranog strojnog učenja* (engl. supervised machine learning) za automatsku indukciju lematizacijskih pravila iz postojećih morfoloških leksikona ili morfološki označenih korpusa. Ovaj pristup osobito je prikladan za morfološki složene jezike koji iziskuju velik broj lematizacijskih pravila. Očiti nedostatak predstavlja potreba za morfološkim leksikonom, razmjerno skupim jezičnim resursom.

Tipičan primjer ovog pristupa jest učenje lematizacijskih pravila slovenskog jezika na temelju morfološkog leksikona [52]. Lematizacijska pravila definirana su kao zamjena sufiksa oblika $X \rightarrow Y$, gdje je X sufiks oblika riječi a Y je sufiks leme. Algoritam učenja koristi paradigmu *slijednog pokrivanja* (engl. sequential covering) kako bi na temelju primjera za učenje oblika $(oblik, lema)$ kombinatoričkom optimizacijom inducirao *ako-onda* pravila zamjene sufiksa. Primjer takvog pravila jest:

if $[ThreeLastCh = hom \vee nom \vee dom]$ **then** $om \rightarrow \varepsilon$.

Kombinatorička optimizacija temelji se na pohlepnom pretraživanju (engl. greedy search) u ovisnosti o kvaliteti pravila mjerenoj na primjerima za učenje. Ovim je pristupom ostvarena točnost lematizacije od oko 60%, odnosno oko 75% kada se primjeni dekompozicija problema metodom *slijednog modeliranja* (engl. sequential modeling).

Sličan pristup, no temeljen na paradigmi *ripple-down pravila* (engl. ripple-down rules, RDR) predložen je u [53]. Nešto drugačija je indukcija lematizacijskih pravila iz morfološki označenog korpusa temeljem *induktivnog logičkog programiranja* (engl. inductive logic programming, ILP) [54].

E. Korjenovanje temeljem sličnosti nizova znakova

Korjenovanje temeljeno na pravilima problematično je kod jezika visoke morfološke složenosti, dok je strojno učena lematizacija problematična kod jezika za koje ne postoje morfološki leksikoni. Alternativu predstavljaju metode *nenadziranog strojnog učenja* (engl. unsupervised machine learning) kojima se rječnici ili pravila korjenovanja induciraju automatski iz neoznačenih korpusa. Prednost ovakvih postupaka jest to što su u načelu jezično neovisni.

Tipičan primjer jest grupiranje morfološki povezanih riječi u klase ekvivalencije (korijenske klase) temeljem neke *mjere sličnosti nizova znakova* (engl. string similarity measure) [55]–[57]. Iz dobivenih grupa riječi moguće je zatim konstruirati normalizacijske rječnike ili inducirati pravila korjenovanja. U [55] mjera sličnosti dviju riječi temelji se na broju podudarajućih podnizova fiksne duljine, odnosno *n-grama*. Mjera sličnosti SC (similarity coefficient) definirana je Diceovim koeficijentom na sljedeći način:

$$\mathrm{SC} = \frac{2 \times (\text{broj zajedničkih unikatnih n-grama})}{\text{zbroj unikatnih n-grama u svakom nizu}} .$$

Korištenjem ove mjere provodi se zatim grupiranje temeljeno na minimalnoj udaljenosti (engl. single linkage clusering) u ovisnosti o heuristički postavljenom pragu. Ovaj je pristup uspješno primijenjen na mnoge jezike, uključivo engleski [58] i turski [59]. U [56] opisana je prilagodba pristupa za arapski jezik u kojoj se, radi učestale infiksacije u jeziku i manje prosječne duljine riječi, umjesto Diceovog koeficijenta koristi Jaccardov koeficijent. Pored toga, grupiranje se provodi tek nakon primjene konzervativnog korjenovanja temeljenog na pravilima.

Sličan pristup, primijenjen na bengalski jezik, opisan je u [57]. Ondje je za mjeru sličnosti (odnosno mjeru udaljenosti) predložena, pored ostalih, sljedeća mjera:

$$D = \frac{n - m + 1}{m} \times \sum_{i=m}^{n} \frac{1}{2^{i-m}} ,$$

gdje je m mjesto prvog nepodudaranja nizova znakova (brojano s lijeva nadesno), a $n + 1$ je duljina nizova (kraći od dva niza nadopunjuje se s desna). Intuitivno, mjera nagrađuje dug zajednički prefiks te kažnjava svako daljnje nepodudaranje znakova. Primjerice, za riječi *astronomer* i *astronomically* vrijedi $m = 8$, $n = 13$, pa $D = \frac{6}{8} \times (\frac{1}{2^0} + \cdots + \frac{1}{2^{13-8}}) = 1.4776$. S ciljem dobivanja što kompaktnijih grupa, za grupiranje je korištena maksimalna, a ne minimalna udaljenost (engl. complete linkage clustering). Kako bi se pronašla optimalna vrijednost praga, napravljena je analiza broja grupa u ovisnosti o vrijednosti praga te je kao optimalna uzeta ona vrijednost oko koje se broj grupa stabilizira. Eksperimenti nad engleskim jezikom pokazali su da je korjenovanje provedeno na ovaj način usporedivo s Porterovim algoritmom [34] u smislu učinaka na performanse sustava za pretraživanje. Pristup je uspješno primijenjen i na mađarskom i češkom jeziku [11].

F. Statistički pristupi korjenovanju

Mnogi pristupi korjenovanju temelje se na statističkim informacijama o pojavljivanju i supojavljivanju riječi u korpusu. U literaturi se takvi pristupi često nazivaju *indukcija morfologije* (engl. morphology induction). Pristupi se razlikuju po tome izvode li neko novo znanje iz postojećeg lingvističkog znanja [51], [60] ili indukciju obavljaju bez ikakvog apriornog znanja (engl. knowledge-free) [22], [61]–[63], a također i prema tome ciljaju li iz korpusa izlučiti skupove korijena i afikasa (engl. affix inventories) ili omogućiti punu morfološku analizu riječi.

Jedan od najranijih pristupa statističkom korjenovanju temelji se na tzv. *raznolikosti sljedbenika* (engl. successor variety) [64]. Raznolikost sljedbenika jest broj različitih slova koja se mogu pojaviti nakon nekog prefiksa. Metoda predložena u [64] pronalazi korijen riječi temeljem pretpostavke da se duljinom prefiksa raznolikost sljedbenika smanjuje, da bi na granici korijena naglo porasla.[4] Granicom korijena može se proglasiti slovo čija raznolikost sljedbenika prelazi neki predefinirani prag (metoda *odsijecanja*), slovo za koje je raznolikost sljedbenika veća od prethodnog slova i idućeg slova (metoda *vrha i platoa*) i sl. Pristup temeljen na raznolikosti sljedbenika, u kojemu se koristilo heurističko poboljšanje metode *vrha i platoa*, isproban je u [14] na engleskom i njemačkom jeziku. Na zadatku grupiranja dokumenata postupak se pokazao tek neznatno lošijim od korjenovanja temeljenog na pravilima.

Statistički pristup korjenovanju predstavlja i već spomenuto particioniranja klasa ekvivalencije dobivenih Porterovim algoritmom [51]. Ovaj pristup iskušan je također na klasama ekvivalencije dobivenim pukim grupiranjem riječi koje dijele zajednički prefiks (početna tri slova), što ne iziskuje nikakvo lingvističko znanje. Dobiveni rezultati usporedivi su s Porterovim [34] i Krovetzovim algoritmom [49].

Interesantan je i postupak indukcije pravila derivacijske morfologije temeljem flektivnog morfološkog leksikona [60]. Postupak najprije grupira morfološki srodne lema iz leksikona, a zatim iz dobivenih grupa izlučuje pravila sufiksacije za tvorbu jedne leme iz druge, gradeći na taj način za svaku grupu pripadna derivacijska stabla. Takva se pravila mogu koristiti za derivacijsko korjenovanje, ali i kao pomoć leksikografima pri izradi jezičnih resursa.

Noviji pristupi indukciji morfologije iskušavaju i kombiniraju različite metode nenadziranog strojnog učenja. U [24] opisan je postupak za indukciju pravila odsijecanja sufikasa temeljen na algoritmu *najmanje duljine opisa* (engl. minimum description length, MDL). U [63] opisan je postupak koji koristi *skriveni Markovljev model* (engl. hidden Markov model, HMM) za pronalaženje najvjerojatnijeg korijena riječi. U [22] opisan je postupak izlučivanja skupa korijena i sufikasa temeljen na minimizaciji broja elemenata u tim skupovima uporabom genetičkog algoritma. U [62] opisan je postupak koji kombinira više vrsta lingvističkoga znanja – ortografiju, semantiku i sintaksu – te koristi analizu raznolikosti sljedbenika i latentnu semantičku analizu (engl. latent semantic analysis, LSA) kako bi iz korpusa izlučio klase ekvivalencija ne samo morfološki već i semantički srodnih riječi.

[4]Metoda je naslijeđena iz strukturalne lingvistike u kojoj se granice morfema nastoje utvrditi temeljem distribucije fonema [25].

IV. Načini vrednovanja

Morfološku normalizaciju može se vrednovati na dva načina: *ekstrinzično* ili *intrinzično*. Ekstrinzično vrednovanje provodi se neizravno, mjerenjem konkretnog učinka morfološke normalizacije na performansu sustava za pretraživanje informacija ili klasifikaciju teksta. Intrinzičnim vrednovanjem izravno se mjeri točnost samog postupka normalizacije neovisno o konkretnoj primjeni.

A. Ekstrinzično vrednovanje

Vrednovanje performansi sustava za pretraživanje uobičajeno se provodi nad standardiziranim ispitnim kolekcijama. Prva kolekcija te vrste je kolekcija *Cranfield* (sažeci znanstvenih članaka), a suvremene kolekcije su TREC (jednojezične ispitne kolekcije za više jezika) [65] i CLEF (višejezične kolekcije novinskih članaka) [66]. Takve se kolekcije sastoje od skupa dokumenata, skupa tema odnosno upita te *ocjena relevantnosti* (engl. relevance judgments) za svaki upit. Ocjene relevantnosti su binarne (dokument je relevantan za upit ili to nije) i donose ih ljudski suci sukladno strogim naputcima. Budući da je kod suvremenih kolekcija zbog velikog broja dokumenata praktički nemoguće napraviti potpune ocjene relevantnosti (ocjene relevantnosti za svaki dokument), koristi se tehnika *objedinjavanja* (engl. pooling) [65] kako bi se dobile relativne ocjene relevantnosti: za zadane upite dohvaćaju se dokumenti uporabom različitih sustava za pretraživanje, objedinjuje se prvih k rangiranih rezultata svakog sustava, a zatim se ocjenjivanje provodi samo nad objedinjenim rezultatima. Ovaj način izgradnje ispitnih kolekcija eksperimentalno je opravdan u [67].

Nad ispitnim kolekcijama performansa sustava za pretraživanje informacija uobičajeno se vrednuje mjerama *preciznosti* P i *odziva* (engl. recall) R, definiranih na sljedeći način [68]:

$$P = \frac{\#(\text{dohvaćeni relevantni dokumenti})}{\#(\text{dohvaćeni dokumenti})} ,$$

$$R = \frac{\#(\text{dohvaćeni relevantni dokumenti})}{\#(\text{relevantni dokumenti})} .$$

Preciznost i odziv međusobno su suprotstavljeni: odziv raste s brojem dohvaćenih dokumenata, dok preciznost tipično pada. Standardna mjera koja na neki način objedinjuje informaciju o preciznosti i odzivu rangiranih rezultata je *srednja prosječna preciznost* (engl. mean average precision, MAP): za svaki upit izračunava se srednja vrijednost preciznosti nakon što je dohvaćen svaki relevantni dokument, a zatim se izračunava prosjek za sve upite. Međutim, za neke primjene, kao što su internetske tražilice, mjera MAP nije indikativna. Za takve je primjene puno prikladnija mjera preciznosti pri fiksnim razinama odziva, tzv. *preciznost na* k (engl. precision at k), često označavana kao $P@k$.

Utjecaj morfološke normalizacije na sustave za pretraživanje informacija najčešće je vrednovan u smislu promjene preciznosti pri fiksnom odzivu (primjerice $P@25$, $P@50$, $P@75$ ili $P@5$, $P@10$, $P@20$) ili promjene MAP vrijednosti [6]–[9], [40], [49], [51]. Uobičajeno se provodi i statistička analiza rezultata kako bi se utvrdilo je li razlika u preciznosti statistički signifikantna. U pogledu odabira najprikladnijeg statističkog testa ne postoji usuglašeno stajalište [4]; ovisno o hipotezi i pretpostavkama o distribuciji podataka, primjenjuje se upareni t-test, Wilcoxonov test, dvofaktorska ANOVA (analiza varijance) ili *samodopunjujuća* (engl. bootstrap) metoda [69], [70].

Vrednovanja utjecaja morfološke normalizacije na klasifikaciju teksta također se temelji na mjerama preciznosti i odziva, zatim na mjeri *točnosti* (engl. accuracy) te mjeri F_1 koja je definirana kao harmonijska sredina preciznosti i odziva, $F_1 = (2PR)/(P + R)$. Standardne ispitne kolekcije su *Reuters-21578* i noviji *Reuters-RCV1* (novinski članci) te kolekcija *20 Newsgroups* (članci iz diskusijskih grupa).

B. Intrinzično vrednovanje

S obzirom da je krajnja svrha morfološke normalizacije poboljšanje performanse sustava za pretraživanje informacija, odnosno sustava klasifikacije teksta, ekstrinzično vrednovanje ne samo da je korisno nego je i neizostavno. Međutim, kao što je istaknuto u [71], ekstrinzično vrednovanje ne pruža baš nikakav uvid u funkcioniranje postupka normalizacije. Točnije, ekstrinzično vrednovanje ne omogućava nam razlučiti između slučajeva neispravne normalizacije od slučajeva u kojima je normalizacija ispravna, ali nije od koristi. Za to je potrebno provesti intrinzično vrednovanje koje će dati ocjenu točnosti normalizacijskog postupka neovisno o nekom konkretnom zadatku. Takvi uvidi mogu biti ključni pri izgradnji i optimizaciji postupka normalizacije, osobito onih temeljenih na pravilima.

U [71] (slično i u [39]) predložen je intrinzičan način vrednovanja temeljen na prebrojavanju pogrešaka *podkorjenovanja* (engl. understemming) i *prekorjenovanja* (engl. overstemming). Pogreška podkorjenovanja nastupa kada dva oblika riječi, premda morfološki povezani, nisu svedeni na zajedničku normu. Pogreška prekorjenovanja nastupa kada se na istu normu svedu dva oblika koja nisu morfološki povezana. Ove se pogreške izračunavaju na ručno sastavljenom uzorku koji se sastoji od grupa morfološki povezanih oblika riječi. Točna normalizacijska procedura na takvom bi uzorku trebala počinjavati što manji broj pogrešaka podkorjenovanja i prekorjenovanja. Indeks podkorjenovanja (UI) i indeks prekorjenovanja (OI) izračunavaju se na sljedeći način:

$$UI = \frac{\text{\#(različito normalizirani parovi u svakoj grupi)}}{\text{\#(parovi riječi svakoj grupi)}} ,$$

$$OI = \frac{\text{\#(parovi iz različitih grupa svedeni na istu normu)}}{\text{\#(parovi svedeni na istu normu)}} .$$

U praksi su ova dva indeksa međusobno povezana: npr. konzervativno korjenovanje rezultirat će s malo pogrešaka prekorjenovanja i puno pogrešaka podkorjenovanja, dok će kod agresivnog korjenovanja situacija biti upravo obrnuta. Ovi se odnosi mogu izraziti *težinom korjenovanja* (engl. stemming weight) definiranom kao $SW = OI/UI$. U [71] je temeljem ove mjere Porterov algoritam [34] ocijenjen kao poprilično konzervativan, Lovinsov [26] kao umjeren, a Paice-Huskov kao agresivan [35]. Interesantno je da kod ekstrinzičnog vrednovanja ovih algoritama nisu ustanovljene nikakve signifikantne razlike [25].

Opisana metoda može se koristiti za vrednovanje točnosti ne samo flektivnih već i derivacijskih normalizacijskih postupka. Međutim, budući da derivacijski povezane riječi ne moraju nužno biti i semantički povezane, odnosno budući da ta veza može biti različite snage i kontekstno ovisna, to se javlja problem kako napraviti odgovarajuće grupiranje. U [71] predloženo je da se grupiranje čini odvojeno na dvije razine, ovisno o snazi semantičke povezanosti. Unatoč tome, čini se da je grupiranje u konačnici ipak u velikoj mjeri proizvoljno.

Lematizacijski postupci mogu se, osim navedenom metodom, intrinzično vrednovati i temeljem usporedbe s postojećim morfološkim leksikonima. U tom slučaju točnost normalizacije može se izraziti u terminima preciznosti, odziva ili mjere F_1. Ovaj postupak međutim inzistira na lingvističkoj točnosti leme, stoga ne mora biti indikativan za točnost normalizacije.

V. ZAKLJUČAK

Morfološka normalizacija sažimlje različite morfološke varijante jedne riječi na jedan jedinstveni oblik. Normalizacijom se uklanjaju negativni učinci morfološke varijacije na pretraživanje informacija i klasifikaciju teksta, ponajviše smanjenje preciznosti pretraživanja pri niskim odzivima te povećanje dimenzije prostora značajki. Potreba za morfološkom normalizacijom osobito je izražena kod morfološki složenih jezika.

U ovome radu načinjena je osnovna podjela pristupa morfološkoj normalizaciji te je dan pregled nekih tipičnih postupaka. Postupci normalizacije razlikuju se, pored ostalog, prema vrsti i količini znanja koja je u njih ugrađena te stupnju automatiziranosti. Svaki od opisanih postupaka ima svoje prednosti i nedostatke. Opisana su i dva različita standardna pristupa vrednovanju postupaka normalizacije: mjerenje učinaka normalizacije na sustav za pretraživanje odnosno klasifikaciju te izravno mjerenje točnosti normalizacije neovisno o zadatku. Prvi način vrednovanja je neizostavan, a drugi je koristan pri izradi i optimizaciji normalizacijskog postupka.

Morfološka normalizacija i dalje je predmetom intenzivnog istraživanja. Veliki izazov predstavljaju mali jezici za koje ne postoje odgovarajući jezični resursi. Za takve se jezike posebno prikladnim čine automatski pristupi korjenovanju temeljeni na metodama nenadziranog strojnog učenja. Zanimljiv je također odnos između derivacijske morfologije i leksičke semantike, koji u kontekstu pretraživanje informacija još uvijek nije dovoljno istražen. Nedovoljno pažnje posvećeno je i problemu homografije, vrlo prisutnom kod morfološki složenih jezika kao što je hrvatski.

LITERATURA

[1] G. Salton, A. Wong, and C. S. Yang, "A vector space model for automatic indexing," *Commun. ACM*, vol. 18, no. 11, pp. 613–620, 1975.

[2] Y. Yang and J. O. Pedersen, "A comparative study on feature selection in text categorization," in *Proceedings of ICML-97, 14th International Conference on Machine Learning*, D. H. Fisher, Ed. Nashville, US: Morgan Kaufmann Publishers, San Francisco, US, 1997, pp. 412–420. [Online]. Available: citeseer.ist.psu.edu/yang97comparative.html

[3] C. Galvez, F. de Moya-Anegón, and V. H. Solana, "Term conflation methods in information retrieval: Non-linguistic and linguistic approaches," *Journal of Documentation*, vol. 61, no. 4, pp. 520–547, 2005.

[4] W. Kraaij and R. Pohlmann, "Viewing stemming as recall enhancement," in *Proc. of SIGIR '96*, 1996, pp. 40–48. [Online]. Available: citeseer.ist.psu.edu/kraaij96viewing.html

[5] A. Pirkola, "Morphological typology of languages for IR," *Journal of Documentation*, vol. 57, no. 3, pp. 330–348, 2001.

[6] D. Harman, "How effective is suffixing?" *Journal of the American Society for Information Science*, vol. 42, no. 1, pp. 7–15, 1991.

[7] D. A. Hull, "Stemming algorithms: A case study for detailed evaluation," *Journal of the American Society of Information Science*, vol. 47, no. 1, pp. 70–84, 1996. [Online]. Available: citeseer.ist.psu.edu/hull96stemming.html

[8] J. Savoy, "Light stemming approaches for the French, Portuguese, German and Hungarian languages," in *SAC '06: Proceedings of the 2006 ACM symposium on Applied computing*. New York, NY, USA: ACM Press, 2006, pp. 1031–1035.

[9] S. Tomlinson, "Lexical and algorithmic stemming compared for 9 European languages with Hummingbird SearchServer at CLEF 2003," in *CLEF*, 2003, pp. 286–300.

[10] M. Popovic and P. Willett, "The effectiveness of stemming for natural-language access to Slovene textual data," *Journal of the American Society for Information Science*, vol. 43, no. 5, pp. 384–390, 1992.

[11] P. Majumder, M. Mitra, and D. Pal, "Hungarian and czech stemming using YASS," in *Working Notes for the CLEF 2007 Workshop*, 2007.

[12] M. Tadić and B. Bekavac, "Inflectionally sensitive web search in Croatian using Croatian lemmatization server," in *Proceedings of 26th International Conference on Information Technology Interfaces (ITI'06)*, V. Lužar-Stiffler and V. H. Dobrić, Eds. SRCE, Zagreb, 2006, pp. 481–486.

[13] D. Lauc, T. Lauc, D. Boras, and S. Ristov, "Developing text retrieval system using robust morphological parsing," in *Proceedings of 20th International Conference on Information Technology Interfaces (ITI'98)*, V. H.-D. Damir Kalpić, Ed. SRCE, Zagreb, 1998, pp. 61–65.

[14] B. Stein and M. Potthast, "Putting successor variety stemming to work," in *Advances in Data Analysis: Selected Papers from the 30th Annual Conference of the German Classification Society*. Springer, 2007, pp. 367–374.

[15] E. Leopold and J. Kindermann, "Text categorization with support vector machines: how to represent texts in input space?" *Machine Learning*, vol. 46, pp. 423–222, 2002.

[16] T. Gaustad and G. Bouma, "Accurate stemming of Dutch for text classification," 2002. [Online]. Available: citeseer.ist.psu.edu/gaustad02accurate.html

[17] M. Malenica, T. Šmuc, J. Šnajder, and B. Dalbelo Bašić, "Language morphology offset: Text classification on a Croatian-English parallel corpus," *Information Processing and Management*, vol. 41, no. 1, pp. 325–339, 2008, doi:10.1016/j.ipm.2006.12.007.

[18] B. Dalbelo Bašić, B. Bereček, and A. Cvitaš, "Mining textual data in Croatian," in *Proceedings of the 28th International Conference MIPRO 2005, Business Intelligence Systems*, 2005, pp. 61–66.

[19] A. Šilić, J.-H. Chauchat, B. Dalbelo Bašić, and A. Morin, "N-grams and morphological normalization in text classification: A comparison on a Croatian-English parallel corpus," in *Lecture Notes in Artificial Intelligence*, vol. 4874. Springer, 2007, pp. 671–682.

[20] C. Liao, S. Alpha, and P. Dixon, "Feature preparation in text categorization," in *Proceedings of Autralasian Data Mining Workshop*, Canberra, Australia, 2003.

[21] F. Sebastiani, "Machine learning in automated text categorization," *ACM Computing Surveys*, vol. 34, no. 1, pp. 1–47, 2002. [Online]. Available: citeseer.ist.psu.edu/sebastiani02machine.html

[22] A. Gelbukh, M. Alexandrov, and S.-Y. Han, "Detecting inflection patterns in natural language by minimization of morphological model," *Progress in Pattern Recognition, Image Analysis and Applications, LNCS*, vol. 3287, pp. 432–438, 2004.

[23] C. Galvez and F. Moya-Anegón, "An evaluation of conflation accuracy using finite-state transducers," *Journal of Documentation*, vol. 62, no. 3, 2006.

[24] J. Goldsmith, "Automatic language-specific stemming in information retrieval," *Revised Papers from the Workshop of Cross-Language Evaluation Forum on Cross-Language Information Retrieval and Evaluation, LNCS*, vol. 2069, pp. 273–284, 2000.

[25] R. Baeza-Yates, *Information Retrieval: Data Structures & Algorithms*. Prentice-Hall, 1992.

[26] J. B. Lovins, "Development of a stemming algorithm," *Translation and Computational Linguistics*, vol. 11, no. 1, pp. 22–31, 1968.

[27] M. Tadić and S. Fulgosi, "Building the Croatian morphological lexicon," in *Proceedings of EACL'2003*, 2003, pp. 41–46.

[28] L. Clement, B. Sagot, and B. Lang, "Morphology based Automatic acquisition of large-coverage lexica," in *Proceedings of LREC'04*, May 2004, pp. 1841–1844.

[29] B. Sagot, "Automatic acquisition of a Slovak lexicon from a raw corpus," *Lecture Notes in Computer Science*, vol. 3658, pp. 156–163, 2005.

[30] A. Oliver, "Use of internet for augmenting coverage in a lexical acquisition system from raw corpora," in *Workshop on Information Extraction for Slavonic and Other Central and Eastern European Languages (IESL 2003), RANLP*, 2003.

[31] M. Forsberg, H. Hammarström, and A. Ranta, "Morphological lexicon extraction from raw text data." in *FinTAL*, 2006, pp. 488–499.

[32] J. Šnajder, B. Dalbelo Bašić, and M. Tadić, "Automatic acquisition of inflectional lexica for morphological normalisation," *Information Processing and Management*, vol. 44, no. 5, pp. 1720–1731, 2008.

[33] L. Karttunen, "Applications of finite-state transducers in natural language processing," *Lecture Notes in Computer Science*, vol. 2088, pp. 34–46, 2001. [Online]. Available: citeseer.ist.psu.edu/409462.html

[34] M. F. Porter, "An algorithm for suffix stripping," *Program*, vol. 14, no. 3, pp. 130–137, July 1980.

[35] C. D. Paice, "Another stemmer," pp. 56–61, 1990.

[36] N. Alemayehu and P. Willett, "Stemming of Amharic words for information retrieval," *Literary and Linguistic Computing*, vol. 17, no. 1, pp. 1–17, 2002.

[37] L. Larkey, L. Ballesteros, and M. Connell, "Improving stemming for Arabic information retrieval: Light stemming and co-occurrence analysis," 2002. [Online]. Available: citeseer.ist.psu.edu/larkey02improving.html

[38] W. Kraaij and R. Pohlmann, "Porter's stemming algorithm for Dutch," 1994. [Online]. Available: citeseer.ist.psu.edu/kraaij94porters.html

[39] W. Kraaij and R. Pohlman, "Evaluation of a Dutch stemming algorithm," 1995. [Online]. Available: citeseer.ist.psu.edu/kraaij95evaluation.html

[40] J. Savoy, "A stemming procedure and stopword list for general French corpora," *Journal of the American Society for Information Science*, vol. 50, no. 10, pp. 944–952, 1999.

[41] T. Kalamboukis, "Suffix stripping with modern Greek," *Program*, vol. 29, no. 3, pp. 313–321, 1995.

[42] R. Schinke, M. Greengrass, A. Robertson, and P. Willett, "A stemming algorithm for Latin text databases," *Journal of Documentation*, vol. 52, pp. 172–187, 1996.

[43] F. Ahmad, M. Yussof, and M. Sembok, "Experiments with a stemming algorithm for Malay words," *Journal of the American Society for Information Science*, vol. 47, no. 1, pp. 909–918, 1996.

[44] A. Kovalenko, "Stemka: Morphological analyser for small search systems," 2002.

[45] D. Weiss, "A survey of freely available polish stemmers and evaluation of their applicability in information retrieval," in *Human Language Technologies as a Challange for Computer Science and Linguistics, Proceedings of the 2nd Language and Technology Conference*, Poznan, Poland, 2005, pp. 216–223.

[46] C. G. Figuerola, R. Gomez, and E. L. de San Roman, "Stemming and n-grams in Spanish: an evaluation of their impact on information retrieval," *Journal of Information Science*, vol. 26, pp. 461–467, 2000.

[47] J. Carlberger, H. Dalianis, M. Hassel, and O. Knutsson, "Improving precision in information retrieval for Swedish using stemming," in *Proceedings of NODALIDA'01*, 2001. [Online]. Available: citeseer.ist.psu.edu/carlberger01improving.html

[48] N. Ljubešić, D. Boras, and O. Kubelka, "Retrieving information in Croatian: Building a simple and efficient rule-based stemmer," in *Digital information and heritage*. Zagreb: Odsjek za informacijske znanosti Filozofskog fakulteta u Zagrebu, 2007, pp. 313–320.

[49] R. Krovetz, "Viewing morphology as an inference process," in *Proceedings of the Sixteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1993, pp. 191–203. [Online]. Available: citeseer.ist.psu.edu/krovetz93viewing.html

[50] M. Lesk, “Automatic sense disambiguation: How to tell a pine cone from an ice cream cone,” in *Proceedings of the 1986 SIGDOC Conference*. ACM, 1986, pp. 24–26.
[51] J. Xu and W. B. Croft, “Corpus-based stemming using cooccurrence of word variants,” *ACM Transactions on Information Systems*, vol. 16, no. 1, pp. 61–81, 1998. [Online]. Available: citeseer.ist.psu.edu/xu98corpusbased.html
[52] D. Mladenić, “Learning word normalization using word suffix and context from unlabeled data,” in *ICML*, 2002, pp. 427–434.
[53] J. Plisson, N. Lavrač, and D. Mladenić, “A rule based approach to word lemmatization,” in *Proceedings of IS-2004*, 2004, pp. 83–86.
[54] S. Džeroski and T. Erjavec, “Learning to lemmatise Slovene words,” in *Learning language in logic, Lecture notes in computer science*, 2000, pp. 69–88.
[55] G. Adamson and J. Boreham, “The use of an association measure based on character structure to identify semantically related pairs of words and document titles,” *Information Processing and Management*, vol. 10, no. 7/8, pp. 253–260, 1974.
[56] A. D. Roeck and W. Al-Fares, “A morphologically sensitive clustering algorithm for identifying arabic roots,” 2000. [Online]. Available: citeseer.ist.psu.edu/deroeck00morphologically.html
[57] P. Majumder, M. Mitra, S. K. Parui, G. Kole, P. Mitra, and K. Datta, “YASS: Yet another suffix stripper,” *ACM Transactions on Information Systems*, vol. 25, no. 4, pp. 18:1–18:20, 2007.
[58] G. E. Freund and P. Willett, “Online identification of word variants and arbitrary truncation searching using a string similarity measure,” *Information Technology: Research and Development*, vol. 1, pp. 177–187, 1982.
[59] F. C. Ekmekcioglu, M. F. Lynch, and P. Willett, “Stemming and n-gram matching for term conflation in Turkish texts,” *Information Research News*, vol. 7, no. 1, pp. 2–6, 1996.
[60] E. G. Xerox, “Unsupervised learning of derivational morphology from inflectional lexicons,” 1999. [Online]. Available: citeseer.ist.psu.edu/237210.html
[61] J. Goldsmith, “Unsupervised learning of the morphology of a natural language,” *Computational Linguistics*, vol. 27, pp. 153–198, 2001.
[62] P. Schone and D. Jurafsky, “Knowledge-free induction of inflectional morphologies,” 2001. [Online]. Available: citeseer.ist.psu.edu/schone01knowledgefree.html
[63] M. Melucci and N. Orio, “A novel method for stemmer generation based on hidden Markov models,” in *Proceedings of CIKM’2003*, 2003, pp. 131–138.
[64] M. Hafer and S. Weiss, “Word segmentation by letter successor varieties,” *Information Processing and Management*, vol. 10, no. 11/12, pp. 371–386, 1974.
[65] D. Harman, “Overview of the first TREC conference,” pp. 36–47, 1993.
[66] M. Kluck, “Test collection report for the CLEF 2003 campaign,” 2003.
[67] E. Voorhees, “The philosophy of information retrieval evaluation.” [Online]. Available: citeseer.ist.psu.edu/613279.html
[68] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge University, 2008.
[69] J. Savoy, “Statistical inference in retrieval effectiveness evaluation,” *Information Processing and Management*, vol. 30, no. 4, pp. 515–533, 1997.
[70] D. A. Hull, “Using statistical testing in the evaluation of retrieval experiments,” in *Research and Development in Information Retrieval*, 1993, pp. 329–338. [Online]. Available: citeseer.ist.psu.edu/hull93using.html
[71] C. D. Paice, “Method for evaluation of stemming algorithms based on error counting,” *Journal of the American Society for Information Science*, vol. 47, no. 8, pp. 632–649, 1996.