Euskararako koma-zuzentzaile automatiko baterantz

Bertol Arrieta, Iñaki Alegria, Arantza Diaz de Ilarraza

Informatika Fakultatea.
Euskal Herriko Unibertsitatea (UPV/EHU)

bertol@ehu.es; i.alegria@ehu.es; a.diazdeilarraza@ehu.es

Jasoa: 2013-05-29 Onartua: 2013-10-28

Laburpena: XUXEN ortografia-zuzentzailearen arrakastaren ondoren eta IXA taldean Hizkuntzaren Prozesamenduan urtetan egindako lanari jarraiki, XUXENg euskarako gramatika- eta estilo-zuzentzailea garatzeko aurrerapausoak egiten dihardugu azken urteetan; horien artean kokatzen dugu hemen aurkeztuko dugun koma-zuzentzaile automatikoa ere. Tresna honen garapenerako, komak zuzen jartzeko lan teorikoak aztertu ditugu lehendabizi, eta ikasketa automatikoko teknikak eta erregeletan oinarritutakoak uztartu ditugu gero; koma-zuzentzaile bat garatzeko sintagmeneta perpausen identifikatzaile automatikoen beharra azaleratu du ikerketa honek.

Abstract: After the success of the Basque spell checker XUXEN, the IXA Natural Language Processing group is working to develop a grammar and style checker, where we include the comma corrector here presented. For this purpose, we first analysed the theoretical works to put commas in Basque, and then we applied both rule based and machine learning based techniques. This work underlines the need of developing both chunk and clause identifiers to develop a comma checker.

1. MOTIBAZIOA: NORK JAN DU AITA?

Gizakiak, garuneko hainbat mekanismoren bidez, gaitasuna dauka hizkuntza —bai idatzia, bai ahozkoa— ulertzeko. Baina guk hain erraz (ia ahaleginik gabe) egiten dugun hori, aldamenekoarekin ahoz edo idatziz komunikatzeko prozesu hori, dirudiena baino zailagoa da; hainbesteraino, ezen hizkuntza —bere osoan— ulertzeko gai den makina bat sortzea ezinezkoa baita gaur egun. Hizkuntzaren anbiguotasun handia da, nagusiki, horren erruduna. Hitz bakar batek, adibidez, hiruzpalau adiera izan ditzake; perpaus baten esanahia ulertzeko, berriz, perpauseko hitz guztien esanahi

egokia bereganatzeaz gain, makinak berdin ulertu beharko lituzke hitzen arteko loturak —sintaxiak adierazten dizkigunak—.-

Zer ulertuko luke makinak, adibidez, «*nork jan du aita?*» esaldia prozesatu behar izango balu? Nola analizatuko luke esaldi hau? Norbaitek aita jan duela pentsatuko luke ziur. Gizakiok, ordea, testuinguruaren arabera esaldi honen benetako zentzua zein den asmatuko genuke. Azken gerezia plateretik desagertu ondoren semeak esandako esaldia dela jakingo bagenu, esaterako, «*nork jan du aita?*» esaldiari «*nork jan du, aita?*» zentzua emango genioke. Hau da, «*nork jan du (azken gerezia), aita?*». Adibide honetan ikus daitekeen moduan, beraz, koma *egokiak* berreskuratzea ezinbestekoa da zenbait kasutan, analizatzaile sintaktiko automatikoak hizkuntzaren anbiguotasun handia ondo ebatz dezan. Antzekoa gertatzen da esaldi honekin ere: «*Haserretu egin zen emaztea beste batekin ikusi zuenean.*». Bere horretan anbiguoa da esaldia, eta koma non jartzen dugun, erabat aldatzen da zentzua:

— *Haserretu egin zen, emaztea beste batekin ikusi zuenean.*
Uler bedi:
Haserretu egin zen (senarra), emaztea beste batekin ikusi zuenean.

— *Haserretu egin zen emaztea, beste batekin ikusi zuenean.*
Uler bedi:
Haserretu egin zen emaztea, (senarra) beste batekin ikusi zuenean.

Bestalde, bi adibide hauek eta antzekoak aztertuz gero, senak diosku sintagmak eta perpausak identifikatu beharko ditugula koma-zuzentzailea garatzeko «*nork jan du aita?*» adibideak erakusten duen moduan, koma, joatekotan, sintagmen artean joango da, eta ez sintagma barruan. Izen-sintagmak eta aditz sintagmak parentesi artean jartzen baditugu, argiago ikusiko dugu: «*(nork) (jan du), (aita)?*»

Adibide honetako izen-sintagmak hitz bakarrekoak badira ere, azterketa sakonagoetan sartu gabe, badirudi sintagmen baitan ez dela komarik joango, oro har.

Bigarren adibideak beste joera bat erakusten digu: komak perpaus baten bukaera edo hasiera adierazten du askotan. Jar ditzagun perpausak eta esaldiak, oraingoan, parentesi artean:

— *(Haserretu egin zen, (emaztea beste batekin ikusi zuenean)).*

— *(Haserretu egin zen emaztea, (beste batekin ikusi zuenean)).*

Begibistan denez, komaren arabera emango zaio zentzu bat edo bestea, eta komaren arabera ikusiko da, era berean, mendeko perpausaren osaera zein den. Hortaz, badirudi komak zuzen jartzea garrantzitsua dela sintag-

meneta perpausen identifikaziorako, eta alderantziz, sintagmeneta perpausen identifikazioa beharrezkoa dela koma-zuzentzaile bat garatzeko. Aurrerago aztertuko dugu gurpil-zoro honi nola egin diogun aurre.

Puntuazio markak euli-gorotzak bezalakoak omen dira: txikiak, beltzbeltzak, ezdeusak; ez omen diegu garrantzirik ematen. Hala zioen Anjel Lertxundi idazleak Berriako bere zutabean. Bere ustez, ordea, puntuazio marka ondo erabiliek morfosintaxiaren ezagutza sakona islatzen dute. Iritzi berekoak dira hizkuntzalari asko ere (Odriozola, 2005; Garzia, 1997; Odriozola eta Zabala, 1993; Nunberg, 1990).

Hala eta guztiz ere, Hizkuntzaren Prozesamenduan (HP) berandu azaleratu zen puntuazioaren garrantzia. Nunbergen monografikoa (Nunberg, 1990) izan zen puntuazioak —eta, zehatzago, komak— HPan izan zezakeen garrantzia mahai gainean jarri zuena. Orduz gero, ugaldu egin ziren puntuazioari buruzko konputazio lanak.

Lan honetan, puntuazio markek eta batez ere komak euskararen prozesamenduan duen eragina aztertu dugu. Era berean, saiatu gara koma-zuzentzaile bat garatzen, horretarako beharrezkoak diren sintagma eta perpaus identifikatzaile automatikoak ere sortuz. Horren guztiaren berri emango dugu artikulu honetan.

2. **TESTUINGURUA**

Esan dugun bezala, euskararako gramatika- eta estilo-zuzentzaile bat garatzeko proiektu orokorragoaren baitan kokatzen da koma-zuzentzailea, eta bere garapena ezin da ulertu IXA taldeak sintaxiaren tratamendu automatikoan egindako lanak kontuan hartu gabe.

2.1. **XUXENg: euskarako gramatika-zuzentzaile automatikoa**

IXA taldean urteak daramatzagu erroreak edo gaizki erabilitako egiturak detektatzen saiatzen. Ortografi zuzentzailea lortzeko helburuarekin egin ziren lehen urratsak (Urkia, 1997; Alegria, 1995; Agirre *et al*., 1992), informazio linguistiko gutxiago behar delako horretarako. Azken urteotan, XUXENg gramatika-zuzentzailea sortzea izan dugu helburu.

Sintaxi-akatsak detektatzea, dena dela, ortografi akatsak detektatzea baino zailagoa da: anbiguotasun handiagoa dago, eta informazio linguistiko gehiago behar izaten da errore horiek detektatzeko. Hala eta guztiz ere, IXA taldean sintaxiaren esparruan lan handia egin da azken urteetan. Besteak beste, euskararen sintaxia lantzeko oinarrizko baliabideak garatu dira (Gojenola, 2000), euskararen desanbiguazio morfologikoa eta azaleko sintaxia landu da (Aduriz eta Diaz de Ilarraza, 2003), euskarako aditzen azpi-

kategorizazioaren azterketa burutu da (Aldezabal, 2004) eta dependentzia-gramatiken formalismoa jarraituz garatutako sintaxi-analizatzailea sortu da (Aranzabe, 2008). Lan hauek oinarri hartuta, gramatika-zuzentzailerako bidean urrats handiak eman dira: Oronozek (2009) postposizio-lokuzio okerrak, data okerrak eta komunztadura-erroreak detektatzeko tesi lana egin zuen, eta Uriak (2009) determinatzaile-erroreak detektatzeko CG[1] erregelak sortu zituen.

Esku artean dugun lana Oronozen (2009) tesi lanaren osagarria dela esan daiteke. Izan ere, erroreen detekziorako berak hizkuntza-ezagutzan oinarritutako hurbilpena erabili bazuen, corpusetan oinarritutakoa erabili dugu guk. Hain zuzen, ikasketa automatikoa baliatu dugu, eta hurbilpen bateko eta besteko teknikak uztartzera ere jo dugu.

2.2. **Sintaxiaren tratamendu automatikoa IXA taldean**

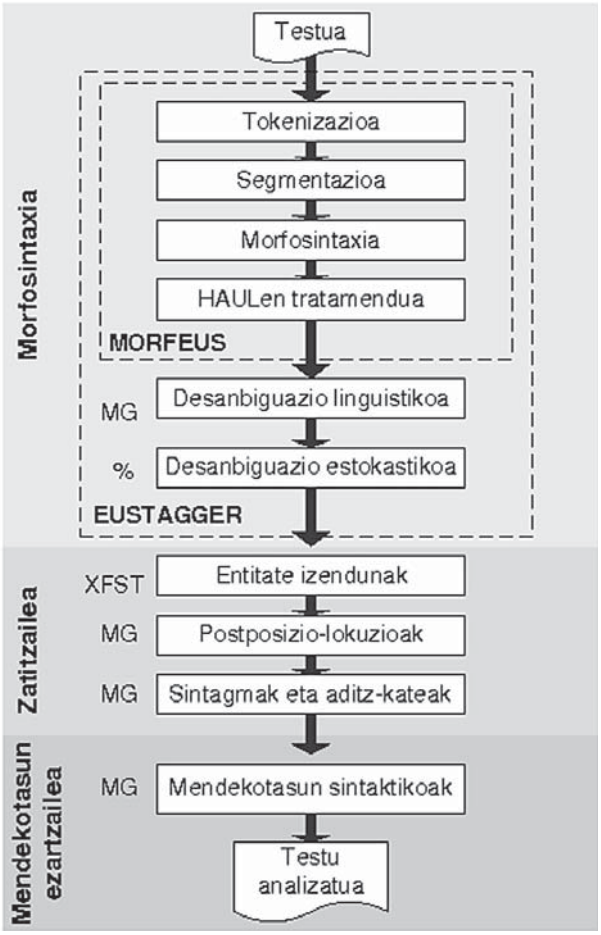
Atal honetan IXA taldean euskararen analisi konputazionalerako erabiltzen diren baliabideak aurkeztuko ditugu: batetik, sintaxi-analisirako sortutako analisi-katea; bestetik, EPEC corpusa.

2.2.1. *Analisi-katea*

Euskararen analisi konputazionalerako, IXA taldean, mendekotasun-edo dependentzia-egituretan oinarritutako sintaxi-analizatzaile sendo bat garatu zen (Aduriz *et al*., 2004). Sintaxi-analizatzaileak geruzaka egiten du lana; geruzetako bakoitzean, hizkuntza ezagutza sakonagoa edo azalekoagoa erabiltzen da, beharraren arabera. Analisi-geruzak katean erabiltzen dira modu sekuentzialean eta moduluetan bilduta. Moduluetako batzuk, mendekotasun-ezartzailea kasu, trukagarriak dira. Analisi-kateko geruza bakoitzak, sarrera moduan, aurreko geruzak eskaintzen dion informazioa erabiltzen du, eta jasotako analisia informazio linguistiko berriarekin aberasten du. Sintaxi-analisia urratsez urrats egiten da honela, eta erabiltzailearen esku geratzen da erabili nahi duen hizkuntza ezagutzaren maila aukeratzea.

Geruzetako bakoitzean bereizketa argia egiten da gramatiken eta gramatika hauek aplikatuko dituzten programen artean. 1 irudian ikus ditzakegu analisi-kateko moduluak eta haien analisi-geruzak. Oronozen (2009) tesi txostenean zehatz-mehatz azalduta datoz analizatzailearen ezaugarriak.

[1] CG: Constraint Grammar formalismoari dagokion akronimoa. Constraint Grammar formalismoak (Karlsson et al., 1995; Tapanainen, 1996) aukera ematen du patroiak identifikatzeko eta etiketak jarri, kendu edo aldatzeko.



1. irudia. Geruza anitzeko euskarako sintaxi-analizatzailea.

Zatitzailean —hau da, sintagmen eta aditz sintagmenidentifikazioan— hobekuntzak egiteko saioak egin dira lan honetan. Horretaz gain, analisi-sintagman agertzen ez den modulu berri bat osatzen saiatu gara: perpausen identifikazioa egiten duen modulua, hain zuzen. Modulu berri horren kokapenaeztabaidagarria da. Batzuek sintagmen identifikazioa baino lehen egitekoa dela uste dute (Tanev eta Mitkov, 2002); besteek, berriz, sintagmeninformazioa darabilte perpaus identifikatzailea hobetzeko: horrela egiten da, hain zuzen, perpausak identifikatzeko antolatu zen ataza partekatuan (Sang eta D'ejean, 2001). Hobekuntza hauek analisi-sintagman txertatzeko, ikasketa automatikoko teknikak baliatu ditugu, eta horiek da-

goeneko garatuta zeuden hizkuntza-ezagutzan oinarritutako gramatikekin uztartu ditugu.

2.2.2. *EPEC corpusa*

EPEC corpusa euskararen prozesamendu automatikorako erreferentzia-corpusa izateko asmoz jaio zen. Euskara batuan idatzitako testuz osatutako corpusa da, eta hainbat mailatan etiketatu da: morfologia eta azaleko sintaxi mailan, lehendabizi, eta sintaxi maila sakonagoan, gero. Corpus honen zati bat *XX. mendeko euskararen corpus estatistikoa*[2] izeneko corpusetik hartu zen; beste zatia, aldiz, *Euskaldunon Egunkaria* berripaperekoa da. Guztira, 200.000 token inguruko corpusa bildu zen.

EPEC corpusak, besteak beste, ikasketa automatikoko teknikak erabiltzeko aukera eman digu.

3. **KOMA-ZUZENTZAILEA**

Atal honetan euskarako koma-zuzentzaile bat garatzeko egindako urratsak azalduko ditugu. Urrats hauek, bai erregela bidezko hurbilpenen bidez, bai ikasketa automatikoko teknikak baliatuz egin dira. Hasteko, ordea, arlo honetako aurrekariak deskriba ditzagun.

3.1. **Aurrekariak**

ACL[3] biltzarrean egindako lantegia da HPan puntuazioarekin zerikusia duten lanen ugalketaren erakusgarri nagusia: *workshop on punctuation in computational linguistics*[4]. Say eta Akmanen (1996) lanak dakar 90. hamarkadaren inguruan egindako lanei buruzko laburpena.

Koma-zuzentzaile edo berreskuratzaile automatikoak sortzeko, berriz, ez dira saiakera asko egin. Hardtek (2001) danierarako koma okerrak detektatzeko saioak egin zituen, Brillen (1995) *transformazioan oinarritutako ikasketa* erabiliz. Horretarako, corpus batean, komak zoriz gehitu zituen lehendabizi. Zoriz gehitutako koma horiek koma oker gisa etiketatu zituen, eta beste guztiak koma zuzen gisa. Modu honetan, koma okerrak detektatzen saiatu zen Hardt (2001). % 91ko doitasuna lortu zuen; estaldura, berriz, % 77koa. Sistema honek ahalmena izango du, neurri batean, gaizki jarritako komak detektatzeko, baina inolaz ez jarri gabe egonik jarri beharko liratekeenak asmatzeko. Gainera, erroreak automatikoki sortzean, errore

[2] www.euskaracorpusa.net

[3] Association for Computational Linguistics.

[4] http://www.hcrc.ed.ac.uk/publications/wp-2.html

oso artifizialak sortzen dira maiz, eta halako sistemek gero eragozpenak izan ohi dituzte errore errealekin.

Baldwin eta Josephek (2009), berriz, puntuazioa (ez soilik komak) berreskuratzeko saioak egin zituzten, ikasketa automatikoko teknikak erabiliz. Zehazki, SVM algoritmoan oinarritutako sailkatzaileen arkitektura bat darabilte, eta F1 = % 62 erdiesten dute. Bestalde, txekierako puntuazioa detektatzeko helburuarekin, sakoneko sintaxi-analizatzaile bat baliatu zuten (Jakubicek eta Horak, 2010) lanean. F1 = % 83,5 lortzen dute ataza horretan, baina eskuz etiketatutako corpus bat erabiliz.

Ahotsaren ezagutzarako sistemek ere puntuazioa berreskuratu behar izaten dute. Shieber eta Taok (2003) osagaien informazioa baliatzen dute, komak non jarri erabakitzeko. Zehatzago esanda, osagaien mugak erabiltzen dituzte komen kokalekua asmatzeko; hau da, token bakoitza zenbat osagairen hasiera eta bukaera den kontatzen dute. Izan ere, euren iritziz, token bat geroz eta osagai gehiagoren muga izan, orduan eta probabilitate handiagoa dago token horren inguruan koma bat izateko. *Markoven eredu ezkutuak (HMM)* erabiltzen dituzte. Lan honen arabera, zentzuzkoa dirudi euskarako sintagmeneta perpausen identifikatzaileek ematen diguten informazioa euskarako koma-zuzentzailea hobetzeko baliatzeak. Bestalde, euren ereduari informazio linguistiko berria gehituz (token bakoitzaren kategoria), komaberreskuratzailearen emaitzak are gehiago hobetu ziren (eskuzko analisiarekin: F1 = % 74,8; analizatzaile automatikoarekin: F1 = % 70,1).

(Shieber eta Tao, 2003) lanaren beste ekarpen garrantzitsua ebaluazioari buruzkoa da. Izan ere, token mailako ebaluazioak emateaz gain, esaldi mailako ebaluazioak ere ematen ditu: esaldi guztiko koma guztiak ondo badaude, esaldia ondo puntuatua izan dela onartuko dugu; komaren bat gaizki badago, ordea, esaldia gaizki puntuatutzat hartuko da. Komaren ebaluaziorako, badu honek zentzua, esaldi beraren barruan koma bat ondo jarri baina hurrengoa gaizki jartzeak, esaterako, esaldi guztiaren zentzua alda baitezake. Lan honetan token mailako ebaluazioa egin bada ere, esaldi mailako azken ebaluazio bat ere egin dugu (ikus 3.3.6 atala).

Bestalde, arestian aipatutako lanetatik, komaren sintaxi-zeregina aztertzen saiatzen direnak ere interesatzen zaizkigu. Izan ere, etorkizunean, komak jartzen *ikasten laguntzen* duen modulu batekin uztartu nahi genuke komen zuzentzailea. Horretarako, ordea, ezinbestekoa da, bai koma non jarri behar den jakitea, baina baita koma leku horretan jartzeko arrazoia ezagutzea ere. Koma bakoitzaren sintaxi-zeregina identifikatu beharko litzateke automatikoki.

(Bayraktar *et al*., 1998) lanean, komaren funtzio edo erabilera desberdinen azterketa egiten da. Delden eta Gomezen (2002) lanean, berriz, ahalegina egiten da koma bakoitzari bere sintaxi-zeregina automatikoki esleitzeko.

3.2. **Komen zuzenketa hizkuntzaren ezagutzan oinarrituta**

Duela urte batzuk, Juan Garziaren[5] komari buruzko teorizazioa (Garzia, 1997) geurera ekartzeko saioak antolatu genituen; beste modu batean esanda, bere teorizazioa nolabait formalizatzeko bilerak egin genituen: informatikaren ikuspuntutik laburtu eta eskematizatu nahi genuen Garziaren komari buruzko teoria. Horretarako, metodologia zikliko bat diseinatu eta bost kideko lan-talde bat osatu genuen (IXA taldeko hiru informatikari eta bi hizkuntzalari). Hala, bilera bakoitzean, adituaren azalpenak entzuten genituen, eta bileraren ostean, azalpen horiek informatikaren ikuspuntutik formalizatzen saiatzen ginen. Komaren arauak zehaztuko zituen erregelen multzoa osatzea zen azken helburua.

Horren ostean, formalismo hori puntuazioaren arloko beste zenbait adituri erakutsi genien (Joxe Ramon Etxebarria[6], Igone Zabala[7] eta Juan Carlos Odriozola[8]). Arestian esan dugun eran, ñabardurak ñabardura, bat zetozen hauek ere Garziarekin formalizatutako komaren arauekin.

Komaren erabilera-arauak finkaturik, koma-zuzentzaile bat lortzeko saioak egin genituen, hizkuntzaren ezagutzan oinarritutako teknikak erabiliz hasieran. Hala, CG formalismoa baliatu genuen, komen arauei zegozkien erregelak idazteko. Gisa honetako erregelak, ordea, testuinguru txikiko arauak formalizatzeko dira egokiak, Oronozen (2009) iritziz. Horregatik, komen arauen artetik CG formalismoarekin inplementatzeko egokiak zirenak soilik aukeratu genituen. Alarma faltsuak ekiditera jo genuen, eta gutxienezko ziurtasun batez detekta genitzakeen arauak soilik inplementatu genituen (doitasun handiagoa bilatu genuen, beraz, estalduraren kaltetan).

2 irudian ikus daiteke erregela baten adibidea, eta (Arrieta, 2010) tesi txostenean ikus daitezke guztiak.

```
MAP (&OKER_KOMA_FALTA_1_1)
                TARGET EDOZEIN_KAT
                IF (1 BAINA + JNT);
```

2. irudia. Komen arauak formalizatzen saiatzeko egindako CG erregelen adibide bat.

[5] Hizkuntzalaria da Garzia, sintaxian eta puntuazioan aditua.
[6] UEUko euskara-zuzentzailea izan zen Joxe Ramon Etxeberria.
[7] Hizkuntzalaritzan doktorea, eta sintaxian eta puntuazioan aditua.
[8] Hizkuntzalaritzan doktorea, eta sintaxian eta puntuazioan aditua.

2 irudiko adibidean dugun erregelak honako hau adierazten du:

Jarri «&OKER_KOMA_FALTA_1_1» etiketa hitz bati —edozein delarik ere bere kategoria—, zeinaren hurrengoa «baina» juntagailua den. Beste modu batean esanda, hitz bat aurkitzen badugu eta bere ondoren datorren hitza «baina» juntagailua bada, tartean koma bat falta da.

Erregela hauek ebaluatzeko, *Euskaldunon Egunkaria*ko garapen corpusa erabili zen (ikasketa automatikoko probak egiteko erabili zen garapen corpus bera, hain zuzen). Erregelek jarritako etiketak testuko jatorrizko komekin alderatuta lortu ziren emaitzak; bitan banatuta aurkezten dira: 0 klasea (ondoren komarik ez duten tokenak) eta 1 klasea (ondoren koma duten tokenak). Bi klase hauen gaineko ohiko neurriak ematen dira: doitasuna, estaldura eta F1 neurria. 3.3.1 atalean, corpusari eta ebaluazioari buruzko xehetasun gehiago irakur daitezke.

1. taula. Komen identifikazioaren emaitzak, CG formalismoa baliatuz egindako erregelekin.

	0			1		
	Doit.	**Est.**	F_1	**Doit.**	**Est.**	F_1
Hizkuntza-ezagutzan oinarrituta	93,1	96,7	94,9	56,9	27,2	36,8

0 klaseko emaitzak onak dira, espero moduan (ikus 1 taula). 1 klaseko emaitzak onegiak ez diren arren (% 36,8ko F1 neurria lortu genuen), esan beharrekoa da arau guztientzat ez ditugula erregelak egin, eta, beraz, estalduraren emaitzak logikoak direla. Bestalde, doitasunari erreparatuta (% 56,9), erregela bidez jartzen diren kometatik, erdia baino gehiago ondo leudeke. Ikasketa automatikoarekin uztartu eta azken emaitza hobetzen laguntzeko moduko erregela multzoa geneukala ebatzi genuen.

3.3. **Komen zuzenketa ikasketa automatikoan oinarrituta**

Jarraian deskribatuko ditugun probetan, corpusetan oinarritutako hurbilpenak erabili genituen koma-zuzentzailea garatzeko; zehatzago esanda, ikasketa automatikoko teknikak. Kasu honetan, hitz bakoitzaren ondoren koma jarri behar den (1) edo ez (0), horixe da ikasi beharreko kontzeptua. Instantziak edo adibideak, berriz, *Euskaldunon Egunkaria*[9] corpusetik lortu genituen. Ikasketa prozesurako, gainera, hainbat ezaugarri linguistiko baliatu genituen. Informazio linguistikoa lortzeko, *Eustagger* erabili genuen,

[9] Euskaldunon Egunkaria eta Berria (www.berria.info) egunkariekin IXA taldeak daukan elkarlanari esker lortutako corpusa.

IXA taldearen analizatzaile/desanbiguatzaile morfosintaktikoa. Honek komak ere erabiltzen ditu, ahalik eta analisi onenak lortu ahal izateko; komak darabiltzan analizatzailea erabiltzearen egokitasuna zalantzazkoa da, ordea, gerora analizatzaile honek ematen duen informazio linguistikoa komazuzentzailea sortzeko erabili behar bada.

Dena dela, hasierako saioak ohiko analizatzailearen bidez egin genituen (komak darabiltzanarekin, alegia).

3.3.1. *Esperimentuen prestaketa*

Atal honetan, corpusaren aukeraketa, ebaluazio-moduaren azalpena, *oinarrizko neurri*ak zein izan ziren, baliatu genituen ikasketa-algoritmoak eta ikasketan —hasiera batean— erabilitako ezaugarri linguistikoak azalduko ditugu, besteak beste.

Corpusaren aukeraketa

Proba gehienak egiteko, *Euskaldunon Egunkaria* berripapereko testuez osatutako corpusa baliatu genuen. 135.000 hitzez osatutako corpusa erabili genuen proba gehienetan; hala eta guztiz ere, corpus handiagoarekin ere egin genituen saio batzuk (ikus 3.3.2 atala). Corpus honek, handia izateaz gain, beste dohain garrantzitsu bat dauka: hasiera batean behintzat pentsatu behar dugu bertako komak arestian aipatutako irizpideei jarraiki jarri zirela; izan ere, egunkariaren estilo-liburuan azaltzen diren komari buruzko arauak bat datoz gureekin, Garziaren (1997) jarraibideak betetzen baitituzte. Corpus honetako testu multzo bat gainbegiratu eta hala zela egiaztatu genuen.

Ebaluazioa

Aurreko atalean deskribatutako neurri estandar berberak erabili ziren: doitasuna, estaldura eta F1 neurria, garapen corpusaren gainean kalkulatuak lehendabizi, eta test corpusaren gainean gero.

Euskaldunon Egunkaria corpusaren 135.000 tokenak zoriz banatu ziren ikasketa eta ebaluazioa egiteko. Horietatik, % 75 ikasketa corpus gisa eta *cross-validation* probak egiteko (ikasketa corpusa hamar zatitan banatuz); gainerako % 25a, berriz, garapen eta test corpus gisa (ikus 2 taula). *Euskaldunon Egunkaria*ko corpus hau erabili zen ia proba guztietan. Beste corpusekin egindako ebaluazioak *cross-validation* teknika baliatuz egin ziren (corpusa 10 zatitan banatuz).

Ebaluazio modu honetan, esan dugun moduan, corpusean jarritako komak baizik ez dira ontzat ematen. Honek baditu bere mugak. Izan ere, ez dakigu ikasten ari garena zenbateraino zuzena den. Gainera, esaldi batean komak jartzeko konbinazio zuzen posible bat baino gehiago egon daitezke,

2. taula. Komak ikasteko eta ebaluatzeko erabilitako *Euskaldunon Egunkaria*ko corpusaren banaketa.

	Token kopurua
Ikasketa-corpusa	101.250
Garapen-corpusa	28.500
Test-corpusa	5.250
Corpus osoa	135.000

eta guk zuzentzat emandakoa —testuen egileek jarritakoa— aukera bat baino ez da. HPko beste zenbait alorretan ere gertatzen den arazo hau aintzat hartuta (Mayor *et al*., 2009), erabaki genuen ebaluazio kualitatibo bat egitea (ikus 3.3.6 atala), aukera bat baino gehiago ontzat emanez.

Azken testa, eskuzko etiketatzea —hizkuntzalariek eginikoa— eta ebaluazio kualitatiboa egiteko, 5.500 hitzeko test corpus txikiagoa erabili genuen.

Ikasi beharreko kontzeptua

Emaitza-atributua edo ikasi beharreko kontzeptua bitarra da kasu honetan; alegia, 0 edo 1 balioak soilik har ditzake. 0 balioak esan nahi du adibide edo instantzia horren ondoren ez datorrela komarik eta 1 balioak, aldiz, koma datorrela adibide horren ostean. Beste hitz batzuetan esanda, garapen edo test corpuseko token bakoitzari zein balio dagokion erabaki behar du sailkatzaileak: token bakoitza 0 klasekoa den edo 1 klasekoa den.

Horretarako, corpusa prestatu behar izan genuen. Komak, hain zuzen, ez ziren adibide edo instantzia gisa gehitu, aurreko tokenaren emaitza-atributu gisa baizik. Hau da, token baten ondoren koma bat baldin badator, emaitza-atributuan 1 balioa izango du token horri dagokion adibideak; bestela, 0 balioa.

Oinarrizko neurriak

Hainbat modu baliatu genituen oinarrizko neurriak kalkulatzeko. Onenak honako hau egiten zuen: ikasketa corpusean komaz jarraituak maizen agertzen diren 200 hitzak hartu, eta garapen corpuseko hitz horien agerpen guztiei koma jarri.

3 taulan ikus daitezke *baseline* delakoarekin lortutako emaitzak. Emaitza onak lortzen dira 0 klaserako; hau da, sistemak ondo ikasten du komak noiz ez diren jarri behar. Jarri behar diren komak jartzen, ordea, ez du batere ondo asmatzen (ikus 1 klaseko emaitzak). Bestalde, oso alde handia

3. taula. *Baseline-neurriak* edo *oinarrizko neurriak.*

	0			1		
	Doit.	**Est.**	F_1	**Doit.**	**Est.**	F_1
baseline_200	94,4	75,6	84,0	12,1	42,7	**18,9**

dago bi klaseen (0,1) arteko emaitzen artean. Izan ere, corpusa ez da *orekatua* zentzu horretan: askoz adibide gehiago ditu 0 klasekoak, 1 klasekoak baino.

Gogoan izan aztertu dugun corpusean token guztien % 8a komak direla.

Alegia, atzean koma duten tokenak askoz gutxiago dira ikasketa corpusean, komaz gabekoak baino. Desoreka hori eta 1 klaseko oinarrizko neurri kaskar horiek egonda, aurreikusi genuen eragozpenak izango genituela klase horretarako emaitza onak lortzeko.

Ikasketa-algoritmoak

Hiru ikasketa-algoritmo hauen WEKA inplementazioak erabili genituen: *Naive Bayes*, erabaki-zuhaitzak (C4.5 algoritmoa) eta *Support Vector Machine* (SVM).

Naive Bayes erabili genuen, algoritmo sinpleenetako bat delako berau; erabaki-zuhaitzak, berriz, morfosintaxiari dagozkion atazetan emaitza onak lortu izan dituelako eta lortzen den ezagutza interpretagarria delako; SVM erabili genuen (C=1), gaur egun gehientsuen erabiltzen den ikasketa-algoritmoa delako eta HPko atazatan emaitza onak erdietsi ohi dituelako.

Atributuak edo ezaugarri linguistikoak

Adibide bakoitzerako —token bakoitzerako, gure kasuan— baliagarriak iruditu zitzaizkigun ezaugarri linguistikoak aukeratu genituen, komari buruz egindako teorizazioa aintzat hartuta. Hala, hasiera batean, erabaki genuen 33 atributu kontuan hartzea; *Eustaggerrek* emandako datuak dira horietako asko (morofosintaxi-ezaugarriak: lema, kategoria, deklinabide-kasua…); besteak *Ixati* zatitzaileak emandakoak (aditz-sintagma baten hasiera edo bukaera den, izen-sintagma baten hasiera edo bukaera den…); beste batzuk CG erregelez osatutako perpaus-mugatzaileak emandakoak dira (esaldiaren hasiera edo bukaera den edo perpaus muga bat den), eta badira batzuk daukagun informazioarekin kalkula daitezkeenak, kontaketa erraz batzuen bidez gehienetan (atributu *kalkulatu* deitu diegu hauei); esate baterako, uneko tokenetik esaldiaren hasierara dagoen aditz-sintagma edo izen-sintagma kopurua.

LEIHOA

Token bakoitzaren atributuen artean, komenigarria da inguruko tokenen informazioa ere kontuan hartzea. Leihoak adierazten du, hain zuzen ere, token bakoitzerako inguruko zenbat token hartzen diren kontuan.

Hasierako gure leihoa (–5,+5) izan zen; alegia, token bakoitzerako, token horren aurreko bost tokenen eta ondorengo bosten informazioa hartzen genuen kontuan.

3.3.2. *Egindako saioak*

Gure sistema fintzeko helburuarekin hainbat proba egin genituen.

LEIHOAREN AUKERAKETA

Aplikazio-leihoa erabakitzeko saioak egin genituen lehendabizi.

4. taula. Garapen corpusean kalkulatutako emaitzak, leihoaren arabera (C4.5 algoritmoa erabilita).

	0			**1**		
	Doit.	**Est.**	F_1	**Doit.**	**Est.**	F_1
(-2,+5)	95,6	98,2	96,9	64,8	43,1	51,8
(-3,+5)	95,7	97,9	96,8	62,7	44,1	51,8
(-4,+5)	95,7	98,0	96,8	63,4	44,6	**52,0**
(-5,+5)	95,5	98,1	96,8	63,5	41,7	50,3
(-5,+4)	95,5	98,2	96,8	64,0	41,7	50,5
(-5,+3)	95,6	98,1	96,9	64,3	43,2	51,7
(-5,+2)	95,6	98,2	96,9	**65,0**	42,4	51,4
(-6,+2)	95,6	98,2	96,9	64,5	42,1	50,9
(-6,+3)	95,6	98,2	96,9	64,6	42,6	51,4
(-8,+2)	95,6	98,2	96,9	64,5	42,5	51,3
(-8,+3)	95,6	97,9	96,7	61,5	43,1	50,7
(-8,+8)	95,6	97,8	96,7	60,4	42,2	49,7

4 taulan ikus daitekeen moduan, ez dago alde handirik leihoaren tamainaren arabera. 0 klaserako ez dago ia alderik. 1 klaserako, berriz, F1 neurrirako bataren eta bestearen arteko aldeak 3 puntutik beherakoak dira, eta zazpi leiho daude emaitza onenetik (F1 = % 52) puntu bakar bateko tartean. Antzeko F1 neurria dutenen artetik, doitasun handiena zuena aukeratu

genuen: (–5,+2) leihoa, hain zuzen ere. Alegia, token bakoitzaren ondoren koma doan ala ez erabakitzeko, aurreko bost tokenen eta ondorengo biren informazioa hartu genuen kontuan.

Izan ere, gramatika-zuzentzaileen eta antzeko tresnen erabiltzaileek nahiago dituzte okerrak zuzentzat hartzen dituzten akats informatikoak, zuzenak okertzat hartzen dituztenak baino. Hau da, doitasunak molde honetako zuzentzaileetan garrantzi handiagoa dauka estaldurak baino (Guinovart, 1996).

Ikasketa-algoritmo egokienaren aukeraketa

Erabakitako (–5,+2) leihoarekin, ikasketa-algoritmoa aukeratzeko probak egin genituen ondoren. WEKA paketeko hiru ikasketa-algoritmo probatu genituen: erabaki-zuhaitzak (C4.5 inplementazioan), *Naive Bayes* eta *Support Vector Machine* (ikus emaitzak, 5 taulan).

0 klaserako emaitzak oso antzekoak dira ikasketa-algoritmo guztietarako. 1 klaserako, aldiz, alde handiak daude. Zalantzarik gabe, erabaki-zuhaitzak dira emaitzarik onenak lortzen dituztenak. Hala ere, deigarria da *Support Vector Machine* algoritmoak lortzen duela doitasun onena (% 67,2), baina oso gutxi arriskatuz, estaldurak adierazten digun moduan (% 14,3).

Badirudi askoz ezaugarri gehiago erabili beharko genituzkeela SVM algoritmoaren bidez emaitza onak lortzeko. Matematikako artikuluak sailkatzeko lan batean, hobekuntza adierazgarriak erdietsi zituzten, hain zuzen, SVM algoritmoaren bidez, 500 ezaugarri *soilik* erabiltzetik 20.000 ezaugarri erabiltzera igaro zirenean (Rehurek eta Sojka, 2010).

5. taula. Garapen corpusean ebaluatutako emaitzak, ikasketa-algoritmoaren arabera ((–5,+2) leihoa erabilita).

	0			1		
	Doit.	**Est.**	F_1	**Doit.**	**Est.**	F_1
C4.5	95,6	98,2	96,9	65,2	42,4	**51,4**
Naive Bayes	94,8	95,6	95,2	37,6	33,5	35,5
SVM	93,6	99,4	96,5	**67,2**	14,3	23,6

Naive Bayes algoritmoa izan zen baztertu genuen lehenengoa. Aipatzekoa da, dena dela, algoritmo honekin ere *oinarrizko neurri*ak gainditu genituela[10].

[10] Zenbait atazatan, oinarrizko neurriak kalkulatzeko erabili ohi da Naive Bayes, bere sinpletasunarengatik.

Bestalde, hurrengo probetarako erabaki-zuhaitzak erabiltzea deliberatu genuen, F1 neurrirako emaitza onenak lortzeaz gain, *SVM* algoritmoa baino askoz azkarragoa baita. Hala eta guztiz ere, *SVM* ez genuen erabat baztertu. Izan ere, corpus handiagoarekin, eta batez ere atributu askorekin, emaitza onak lor ditzakeela esaten da literaturan (Milenova *et al*., 2005; Joachims, 1998).

Atributu berrien gehikuntza

Emaitzak uste bezain onak ez zirenez, erabaki genuen informazio berria gehitzea, hots, atributu edo ezaugarri berriak eranstea. Hala, komaren aurretik maizen agertzen diren hitzak atributu bitar gisa gehitu genituen; hau da, gure ikasketa corpusa aztertu genuen komaren aurretik maizen agertzen ziren ehun hitzak, ehun hitz-bikoteak (*bigram*ak) eta ehun hitz-hirukoteak (*trigram*ak) lortzeko eta atributu gisa erabiltzeko (300 atributu berri, guztira).

6 taulan ikus daitezke lortutako emaitzak. 1 klaseko emaitzak nabarmen hobetu ziren.

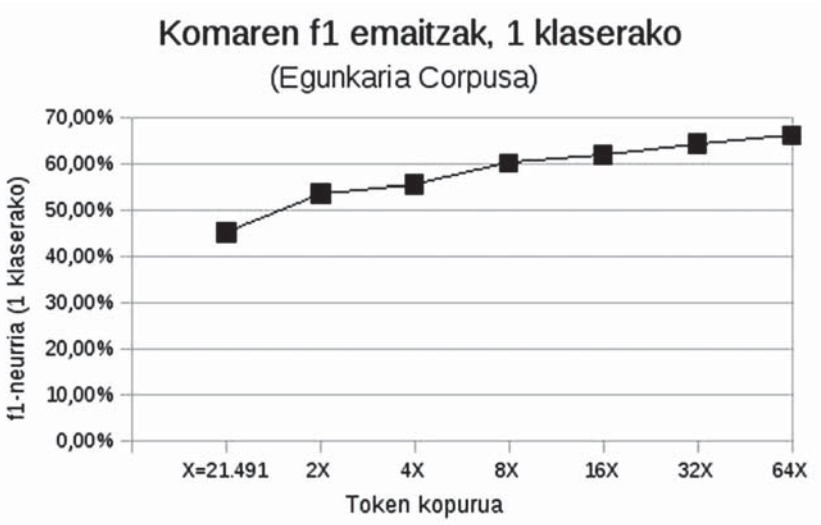
6. taula. Garapen corpusean ebaluatutako emaitzak, atributu berriak gehituta ala gehitu gabe (erabaki-zuhaitzak eta (–5,+2) leihoa erabilita).

	0			1		
	Doit.	**Est.**	F_1	**Doit.**	**Est.**	F_1
Atributu berririk gabe	95,6	98,2	96,9	65,2	42,4	51,4
(1) 300 atributu berri	96,0	98,3	97,2	69,6	48,6	**57,2**

Corpusaren tamainaren eragina

Corpusaren tamainaren eragina ere aztertu nahi izan genuen, corpusa handitzearekin emaitzak zenbat hobetzen ziren ikusteko. Erabaki-zuhaitzak (C4.5) erabili genituen ikasketarako, (–5,+2) leihoa, eta 300 atributu gehigarriak.

3 irudian ikus daitekeen moduan, corpusa gero eta handiagoa izan, emaitzak orduan eta hobeak dira. Gainera, badirudi emaitzak gehiago hobe daitezkeela corpusaren tamaina are gehiago handituta; alegia, corpusaren tamaina handituz lor daitekeen goi mugara ez gara heldu. Dena dela, kontuan hartu behar da proba bakoitzean aurreko proban erabilitako corpusaren bikoitza erabili genuela. 0 klaseko emaitzak ez zaizkigu esanguratsuak iruditu, eta ez ditugu jarri. Baiki, 0 klaseko emaitzak oso onak dira hasiera-hasieratik, eta ez dausintagmahobetzeko tarte handirik. Goi muga kalkulatzeko, dena dela, beste modu bat aurreikusi genuen: puntuazioan adituak diren bi hizkuntzalarik corpus txiki bat eskuz etiketatzea; berek lortutako



3. irudia. Corpusaren tamainaren eragina koma zuzenen identifikaziorako, *Euskaldunon Egunkaria* corpusa baliatuta (hamar zatiko *cross-validation* baliatuta lortutako emaitzak).

emaitza hartuko genuen goi mugatzat (ikus 3.3.6 atala). Hala ere, ikasketan emandako denbora larregi handitzen da corpusa handitzearekin. Beraz, hemendik aurrerako probak ere, orain artekoak bezala, erabaki genuen100.000 bat tokeneko ikasketa corpusarekin egitea.

Sintagmen eta perpausen identifikatzaileen informazioa koma-zuzentzailea hobetzeko

Sintagmeneta perpausen identifikatzaileek ematen duten informazioa balio handikoa iruditzen zaigu koma-zuzentzailerako, are gehiago (Shieber eta Tao, 2003) lanean esandakoa kontuan hartuz gero: osagaien mugei dagokien informazioa erantsiz, beren koma-berreskuratzailea hobetzea lortu zutela, hain zuzen. Intuitiboki ere hala dela esatea ez da zentzugabekeria. Izan ere, arestian ikusi dugun moduan, sintagmabaten barruan ez da oro har komarik izango; bestalde, badira perpaus muga batzuk komaz markatu behar direnak.

Arrazoi hauek direla medio, *FR-Perceptron* algoritmoa (Carreras *et al*., 2005) baliatuz euskarako sintagmaeta perpaus identifikatzaileak garatu genituen. Izan ere, algoritmo honek oso emaitza onak lortzen ditu, oro har azaleko analisiari dagozkion atazetan, eta bereziki ingeleseko perpausen identifikazioan.

FR-Perceptron algoritmoak bi mailatan edo geruzatan dihardu:

— Lehenengoan, hitz mailako iragazketa egiten da (*filtering*): esaldiko *hitz multzo* posible guztiak detektatzen dira, hau da, *hitz multzo hautagaiak*. Beste modu batean esanda, hitz bakoitza *hitz multzo* baten hasiera edo bukaera izan daitekeen ala ez erabakitzen da. Aukeratutako *hitz multzo* hautagai guztiek ez dute zertan koherente izan esaldirako.
— Bigarrenean, *hitz multzo* mailan lan egiten da. Geruza honetan, lehen geruzan iragazitako *hitz multzo hautagaiak* puntuatzen dira (*ranking*), eta esaldirako *hitz multzo*en segida onena aukeratzen da. Alegia, *hitz multzo* hautagai bakoitzari puntuazio bat ematen zaio. Puntuazio horrek adierazten du testuinguru horretan *hitz multzo* hori esaldian zenbateraino den hautagai sendoa.

Esaldiaren azken puntuazioa, beraz, aukeratutako *hitz multzo* hautagaiek duten puntuazioen batura izango da.

Hortaz, hiru ikasketa-funtzio daude guztira: iragazketako *start* eta *end* funtzioak, hurrenez hurren hitz bakoitza *hitz multzo* baten hasiera edo bukaera izan ote daitekeen erabaki beharko dutenak, eta hirugarrenik, *score* deiturikoa, *hitz multzo* hautagai bakoitzari hautagaitzaren sendotasunaren arabera puntuazio bat emango diona. *Pertzeptroi*en algoritmoaren halako orokortze bat baliatzen da hiru ikasketa-funtzioak inplementatzeko.

Euskarako sintagmaeta perpaus identifikatzaileak erdiesteko, hainbat proba egin genituen, baina jarraian azaltzen dira emaitzarik onenak eman zizkiguten konbinazioak.

Euskarako sintagmaidentifikatzailerako, F1= % 83,17 lortu genuen test corpusean, ikasketako automatikoko atributu gisa tokena bera, tokenaren kategoria morfosintaktikoa, deklinabidearen informazioa eta erregeletan oinarritutako sintagmeninformazioa erabiliz. Perpaus identifikatzailean, berriz, F1= % 77,24 erdietsi genuen test corpusean, atributu hauek baliatuz: tokena, lema, kategoria, azpikategoria, deklinabidea, mendekotasunari buruzko informazioa eta erregeletan oinarritutako perpausen informazioa.

Euskarako sintagmaeta perpaus identifikatzaile hauen informazioa lorturik, gure azken asmoa informazio hau atributu gisa eranstea zen, komen zuzentzaile automatikoa garatzeko generabilen ikasketa prozesuan. Arestian esan dugun moduan, ordea, zalantzagarria da komak darabiltzaten sintagmaeta perpaus identifikatzaileak erabiltzearen zilegitasuna, zertarako eta komak ikasteko. Horregatik, sintagmeneta perpausen identifikatzaileak aldatu beharrean izan ginen: sintagmeneta perpausen identifikatzaile berriak sortu behar izan genituen, jatorrizko komak kontuan hartzen ez zituztenak euren ikasketa prozesuan.

Sintagmaeta perpaus identifikatzaile gisa sortutako sailkatzaileei komarik gabeko test corpus bat sartuta, emaitzen beherakada handirik ez eragitea zen helburua. Horretarako, sailkatzaile hauek sortzeko erabilitako ikasketa corpusetik komak kendu, eta berriz sortu genituen sintagmaeta perpaus identifikatzaileak. Komarik ez darabiltzaten hauei, sintagmaeta perpaus identifikatzaile *komagabea*k deitu diegu.

7 eta 8 tauletan ikus daitezke komarik gabeko sintagmeneta perpausen identifikatzaileen emaitzak, aurreko atalean aurkeztutako sintagmeneta perpausen identifikatzaile komadunekin alderatuta (test corpusean egindako ebaluazioa).

7. taula. Komaren eragina, *FR-Perceptron* bidezko euskarako sintagmenidentifikatzailean.

Ikasketa-corpusa	Test-corpusa	Desanb.	$F_1 neurria$
Komaduna	*Komaduna*	autom	83,17
Komagabea	*Komagabea*	autom.	82,24

8. taula. Komaren eragina, *FR-Perceptron* bidezko euskarako perpausen identifikatzailean.

Ikasketa-corpusa	Test-corpusa	Desanb.	$F_1 neurria$
Komaduna	*Komaduna*	autom.	77,24
Komagabea	*Komagabea*	autom.	73,66

Datu hauen arabera, badirudi koma garrantzitsuagoa dela perpausak identifikatzeko, sintagmak identifikatzeko baino.

Komarik gabeko corpusean oinarritutako sintagmaeta perpaus identifikatzaile hauek emandako informazioa erabili genuen komen ikasketa hobetze aldera, emaitzen beherakada ez zitzaigulako handiegia iruditu, batetik, eta gaizki jarritako komen eragin negatiboa saiheste aldera, bestetik.

9 taulan ikus dezakegun eran, 1 klaseko emaitzak zazpi puntu hobetu ziren sintagmaeta perpaus identifikatzaile *komagabe*ek emandako informazioarekin; hamar puntu baino gehiago, sintagmaeta perpaus identifikatzaile komadunekin.

Hobekuntza hauek, gainera, esanguratsuak direla egiaztatu ahal izan genuen, McNemar testa eginez ($p < 0,05$), bi kasuetan. Sintagmeninformazioari esker lortu zen hobekuntza oso txikia den arren, hobekuntza handiak erdietsi ziren perpaus identifikatzaileak emandako informazioa gehituta.

9. taula. Garapen corpuseko emaitzak, *FR-Perceptron* algoritmoaren bidez sortutako sintagmaeta perpaus identifikatzaile *komagabe*ak edo *komadun*ak emandako informazioa gehitu aurretik eta gehitu ondoren (erabaki-zuhaitzak, (–5,+2) leihoa eta 300 atributu gehigarriak erabilita).

	0			1		
	Doit.	Est.	F_1	Doit.	Est.	F_1
Kate-info. eta perpaus-info gabe	96,0	98,3	97,2	69,6	48,6	57,2
Kate-ident *komagabearen* info. gehituta	96,0	98,4	97,2	70,4	48,5	57,4
Kate- eta perpaus-ident *komagabeen* info. gehituta	96,6	98,7	97,6	76,6	55,7	**64,5**
Kate-ident *komadunaren* info. gehituta	96,2	98,5	97,3	73,0	50,7	59,8
Kate- eta perpaus-ident *komadunen* info. gehituta	96,9	98,7	97,8	78,4	59,8	**67,9**

Hiru puntu eta erdiko aldea dago sintagmaeta perpaus identifikatzaile *komadun*ak erabiltzetik *komagabe*ak erabiltzera (% 64,5 vs % 67,9), perpaus identifikatzaile eta sintagmaidentifikatzaile *komadun*ek komagabeen aldean erakusten duten portaera hobea dela eta (ikus 7 eta 8 taulak). Beraz, gure hipotesia betetzen dela baiezta genezake: sintagmeneta perpausen informazioa garrantzitsua da koma-zuzentzailerako. Bestetik, argi dago informazio linguistiko esanguratsua gehitzeak onurak dakartzala, eta informazio linguistiko horrek geroz eta kalitate hobea izan, orduan eta emaitza hobeak lortzen direla.

3.3.3. *Komen zuzenketa, erregelak eta ikasketa automatikoa uztartuz*

Uztartzen baldin baditugu corpusetan oinarritutako teknikak (kasu honetan, ikasketa automatikokoak) eta hizkuntzaren ezagutzan oinarritutakoak (erregela bidezkoak), bataren eta bestearen emaitzak hobetu egiten dira eskuarki. Komarenean ere gauza bera egin genuen: *stacking* teknika erabiliz, 3.2 atalean aurkeztutatako CG erregelek emandakoa informazioa gehitu genion, atributu gisa, ikasketa automatiko bidez lortutako sailkatzaileari.

Sintagmaeta perpaus identifikatzaile *komagabe*ak erabiltzen ditugunean, ikasketa automatikoko teknikak erregelekin uztartzean lortzen diren emaitzak hobegoak dira erregelekin soilik lortutakoak baino; hala ere, ikasketa automatiko hutsarekin lortutakoen oso antzekoak dira (desberdintasuna, hain zuzen, ez da estatistikoki esanguratsua, McNemar testaren arabera; $p < 0, 05$). Izan ere, komak berreskuratzeko egindako CG erregelek oso estaldura apala zuten 1 klasean (% 27,2; ikus 1 taula).

10. taula. Garapen corpusaren gainean ebaluatutako koma-zuzentzailearen emaitzak, hizkuntzaren ezagutzan (CG erregelak) eta corpusetan oinarritutako teknikak (erabaki-zuhaitzak, (–5,+2) leihoarekin eta 300 atributu gehigarriekin) uztartuz, sintagmaeta perpaus identifikatzaile *komadun*arekin edo *komagabe*arekin.

	0			1		
	Doit.	**Est.**	F_1	**Doit.**	**Est.**	F_1
CG erregelak	93,1	96,7	94,9	56,9	27,2	36,8
Ikasketa automatikoa *KPI-komagabearekin*	96,6	98,7	97,6	76,6	55,7	64,5
CG erregelak + ikask. autom. *KPI-komagabearekin*	96,5	98,8	97,6	77,8	55,0	64,4
Ikask. autom. *KPI-komadunarekin*	96,9	98,7	97,8	78,4	59,8	67,9
CG erregelak + ikask. autom. *KPI-komadunarekin*	97,0	98,7	97,8	79,0	61,4	**69,1**

Hala eta guztiz, estatistikoki esanguratsua da sintagma eta perpaus identifikatzaile *komadun*ak darabiltzan ikasketa automatikoko algoritmoak erregela bidezkoekin konbinatzean lortzen den hobekuntza, ($p < 0, 05$), bai erregelekin soilik lortutakoekin konparatuta, bai ikasketa automatikoko teknikekin soilik lortutakoekin erkatuta.

Etorkizunean, dena dela, saio bat egin nahi genuke komak berreskuratzeko erregela multzo osoago bat lortu eta emaitzak modu horretan gehiago hobetu ote ditzakegun aztertzeko.

3.3.4. *Jatorrizko komen eragina saihesten*

Orain arte azaldutako proba guztietan, hainbat tresna erabili genituen informazio linguistikoa lortzeko: *Eustagger* analizatzaile/desanbiguatzaile morfosintaktikoa, *Ixati* zatitzailea, CG erregela bidezko perpaus-mugatzailea eta ikasketa automatikoko sintagmaeta perpaus identifikatzaileak. Erraz uler daiteke tresna hauek jatorrizko komak ere erabiltzen dituztela ahalik eta emaitza onenak lortu ahal izateko, eta horretan datza arazoa: besteak beste, *Eustaggerrek* komak erabiltzen ditu morfosintaxi-analisia lortu eta desanbiguatzeko, eta gero guk informazio linguistiko hori bera erabiltzen dugu koma okerrak detektatu eta zuzenak jartzeko. Jatorrizko komak egokiak baldin badira, lagungarria gerta daiteke koma horiek erabiltzea; alabaina, komak gaizki jarrita baldin badaude (maila baxuko euskara-ikasleen testuekin ari bagara, esaterako), koma hauekin lortzen den informazio linguistikoaren kalitatea okerragoa izango da ziur aski, eta, ondorioz, baita lortuko dugun azken emaitza ere.

Eustaggerrek, batez ere, desanbiguazio prozesuan erabiltzen ditu komak.

Desanbiguazio prozesua bi mailatan egiten da: lehenengoan, CG erregelak erabiltzen dira desanbiguazioa fintzeko; bigarrenean, eredu estokastiko bat erabiltzen da analizatzaileak desanbiguatzen ikas dezan. Erregelei dagokienez, % 11k erabiltzen dute koma (2055 erregeletatik 220 erregelatan baliatzen da komaren informazioa); eredu estokastikoari dagokionez, berriz, ikasketa egiterakoan komadun corpusa baliatzen da. Hori dela eta, bi urrats hauek moldatu behar izan genituen gure analizatzaile berezia inplementatzeko. Batetik, koma zerabilten erregelak baliogabetu behar genituen (baita zatitzailearenak eta mugatzailearenak ere); horretarako, nahikoa izan zen corpusetik komak kentzea[11].

Estokastikoaren kasuan, bestalde, corpus komagabearekin berriz trebatzea zen soluzioa. *Eustaggerrek* darabilen ikasketa corpusari komak kendu eta horrekin entrenatu genuen estokastikoa, eta gero, desanbiguazio prozesuan txertatu genuen.

Bi urrats horiek eman ondoren, prest geneukan *Eustagger komagabe*a: komak zuzentzeko prozesuan erabiltzeko moduko analizatzaile/desanbiguatzaile automatiko berezia, komak ezertarako erabiltzen ez dituena. Ohiko analizatzailearekin —*Eustagger komadun*arekin— alderatuta, *Eustagger komagabe*aren errore tasa % 6 inguru handiagoa da, eta komen ikasketa prozesuan honek eragina izango zuela aurreikusi genuen (ikus 11 taula).

11. taula. Koma-zuzentzailearen emaitzak (garapen corpusean neurtuak), analizatzaile/desanbiguatzaile komadunarekin edo komagabearekin lortutako informazio linguistikoa baliatuz; CG erregelak eta ikasketa automatikoko teknikak uztartuz (sintagmaeta perpaus identifikatzaile komagabeak erabilita).

	0			1		
	Doit.	**Est.**	F_1	**Doit.**	**Est.**	F_1
Corpus komaduna + ***Eustagger komaduna***	96,5	98,8	97,6	77,8	55,0	64,4
Corpus komagabea + ***Eustagger komagabea***	95,0	98,8	96,9	69,3	33,3	45,0

Uste bezala, *Eustagger komagabe*aren portaera okerragoa izateak badu eragina, eta ikasketarako beharrezkoa den informazioa biltzeko egiten diren urratsetako bakoitzak aurrekoaren informazioa darabilenez, ia 20 pun-

[11] Hala eta guztiz ere, aplikatzen ez diren erregela horiek etorkizunean ordezkatzea aurreikusten dugu, desanbiguatzaile, zatitzaile eta mugatzaile ahalik eta onenak izatea komeni zaigulako. Modua aurkitu beharko da, komaren bitartez adierazten zena, beste elementu linguistiko batzuen bidez adierazteko.

tuko galera sortzen du, azkenerako, *Eustagger komagabe*aren errore tasa handiagoak.

Hala ere, komarik gabeko corpusarekin testa egitean, *Eustagger komagabe*arekin emaitza hobeak erdiesten dira, *Eustagger komadun*arekin baino; izan ere, ikasketa automatikoan sarri ikusten den moduan, emaitzek okerrera egiten dute, baldin ikasketarako eta testerako informazio desberdina erabiltzen bada.

3.3.5. *Adibideen azterketa*

Corpus eta analizatzaile/desanbiguatzaile *komadun*ak edota *komagabe*ak erabiliz lortutako koma-zuzentzaileen emaitzak konparatu nahi izan genituen, adibide jakin batzuek eta besteek zeukaten portaera aztertuz.

Lau adibide hauetan 11 taulako bi aukerek adierazten duten portaera aztertuko dugu (aukera bakoitza letra banarekin izendatu dugu, azalpenak errazteko asmoz):

— Corpus komaduna + *Eustagger komadun*a (A aukera).
— Corpus komagabea + *Eustagger komagabe*a (B aukera).

Adibidea 3.1

1. Azken hiru hilabeteetan janaria erosteko dirua bakarrik ematen zietela salatu dute etorkinek, eta euren egoera salatuz gero beren kanporaketa bultzatzeko mehatxua egin zietela enpresaburuek.

2. Ez du, ordea, aipatu beste delinkuentzia mota hau.

3. Besteak beste, Hitchcock, Godard, Wilder eta Stanley Donenen zenbait maisu lan erakutsiko dira bertan.

4. Volker Schlondorff, berriz, Alemaniako zinema garaikidearen bultzatzaile nagusietakoa.

Lehenengo adibidean, dagoen koma zuzen bakarra ondo identifikatu du A aukerak; B aukerak, ordea, ez. Ulergarria da A aukerak koma hori identifikatzea; izan ere, komaren ondoren datorren «eta» hitzean esaldi hasierako etiketa dauka, esaldi eta perpaus mugen CG gramatikak emana, bere informazioaren artean. B aukerak ez du informazio hau, CG gramatika horrek komaren informazioa baliatzen duelako esaldi muga horiek jartzerakoan: hau da, «koma + juntagailua» patroia topatzen duenean, esaldiaren hasieramarka jartzen dio juntagailuari. B aukeran, ordea, ez dugu komarik corpusean, eta beraz esaldi eta perpaus mugen CG gramatikak ez dio etiketa hori jarriko. Hala, zailagoa izango du koma hau identifikatzea.

Bigarren adibidean, dauden bi koma zuzenak identifikatu dituzte bi aukerek, baina B aukerak soberan dagoen bat gehitu du *aipatu* hitzaren

ondoren. Honen arrazoia *aipatu* hitzaren informazioaren artean, perpaus bat hasi eta bukatu dela dioen etiketa izan daitekeela uste dugu, etiketa hau ez baitu A aukerak. Beraz, kasu honetan, *FR-Perceptron* bidezko perpaus identifikatzailearen portaera okerrak eraman du tresna erabaki oker bat hartzera.

Hirugarren adibidean, A aukerak ondo jartzen ditu koma guztiak; B aukerari falta zaio «Godard» eta «Wilder» hitzen arteko koma. Aukera bakoitzak darabilen informazioa aztertuz gero, sintagmenCG gramatikek egindako okerrak *FR-Perceptron* bidezko kate-detektatzailearen portaera okerra dakarrela ikusi dugu, eta honek, segur aski, koma horiek ez identifikatzea.

Laugarren adibidean, ostera, A aukerak bi komak ondo jartzen ditu, ziurrenik *Eustaggerrek* asmatu egin duelako *berriz* hitzaren analisian, eta lokailua dela identifikatu duelako; informazio honekin komen CG gramatikak *berriz* hitzari eta aurreko «Schlondorff» hitzari koma bat dagokiela dioen etiketa esleitu die, eta informazio hau ziurrenik esanguratsua izan da bi hitz hauei koma bana esleitzean. B aukerak, ordea, analizatzaile/desanbiguatzaileak adberbiotzat hartu du «berriz» hitza, eta, beraz, komen CG gramatikak ez ditu A aukerarekin gehitutako etiketak erantsi. Hala eta guztiz ere, B aukerak ondo jarri ditu bi komak.

Adibide hauek aztertu ondoren, atera ditzakegun ondorioak hauek dira: batetik, analizatzaile/desanbiguatzaile bat ala bestea erabilita aldea handiegia ez den arren, lehen urrats honetan gertatzen diren akatsek beste errore batzuk dakartzatela; eta bestetik, analizatzaile/desanbiguatzaileak akatsak egin gabe ere, tarteko urratsetan aplikatzen diren CG gramatikek eta *FR-Perceptron* bidezko algoritmoek erroreak egiten dituztela corpus komagabea erabiltzen badugu. Eta akats batek beste bat dakarrenez, komaren zuzenketan eragina izatea dakar honek azkenerako.

3.3.6. *Ebaluazio kualitatiboa*

Atal honetan, komak ebaluatzeko egin dugun ebaluazio kualitatiboaren emaitzak aztertuko ditugu. *Eustagger komadun*ak, sintagmaeta perpaus identifikatzaile *komagabe*ak eta komak zuzentzeko egindako CG erregelek emandako informazioa gehituta sortutako sailkatzailea ebaluatu genuen.

5.500 tokeneko test corpusa harturik, bi hizkuntzalariri eman genien —aurrez, corpusari, zeuzkan komak kenduta—, eurek komak jar zitzaten. Test corpusak berez zituen komak zuzentzat emanez, bi hizkuntzalarien etiketatzeak test corpusaren komekiko zeukan «bateragarritasuna» neurtu genuen. 12 taulan ikus daitezke ohiko neurriak.

Deigarria da hizkuntzalarien emaitzak % 80tik gertu ibiltzea, eta baten eta bestearen artean sei puntuko aldea egotea.

12. taula. Sailkatzailearen iragarpena eta hizkuntzalarien etiketatzearen emaitzak, test corpuseko jatorrizko komekiko.

	0			1		
	Doit.	Est.	F_1	Doit.	Est.	F_1
Ikask. autom. *KPI-komagabearekin* + CG erregelak	95,6	98,5	97,1	77,6	52,7	62,8
Hizkuntzalari1	98,5	97,6	98,0	79,1	85,9	82,3
Hizkuntzalari2	97,5	97,4	97,5	76,1	76,4	76,3

Bestalde, bi etiketatzaileren arteko adostasuna neurtzeko *kappa-neurri*a erabiltzea gomendatzen da (Carletta, 1996): klase bakoitzaren adibideen kopurua eta banaketa kontuan hartzen du neurri honek. % 74,02koa da bi hizkuntzalarien arteko *kappa-neurria*. *Kappa-neurri*aren balioak interpretatzeko irizpide anitz egonik ere, Carlettak (1996) berak % 80tik gorako balioak jotzen ditu fidagarritzat; % 67 eta % 80 artekoak, berriz, zalantzagarriak direla dio.

Neurri hauez gain, hizkuntzalari bakoitzaren eta sailkatzailearen arteko *kappa-neurriak* kalkulatu genituen; lehenengo hizkuntzalariaren erabakien eta test corpuseko jatorrizko komen arteko *kappa-neurria* % 80,36koa da, eta bigarren hizkuntzalariaren eta test corpuseko jatorrizko komen artekoa, % 73,74koa.

Datu hauek erakusten digute komak berreskuratzearen ataza ez dela batere erraza. Bi hizkuntzalarien test corpusarekiko bateragarritasun maila eta hizkuntzalari batek bestearekiko duena ikusita, 1 klasearen goi muga % 76 ingurukoa dela esatera ausartuko ginateke. Hau da, koma-zuzentzaile automatiko baten *skyline* edo goi muga % 76 ingurukoa dela uste dugu.

Hala eta guztiz ere, azken proba gisa, gure sailkatzailearen ebaluazio osoago bat egitea erabaki genuen. Izan ere, orain arte erabilitako neurriek ez dute adierazten komak automatikoki berreskuratzeko gaitasuna, corpuseko jatorrizko komekiko zuzentasuna baizik; hau da, egileak jarri dituen komekiko bateragarritasuna adierazten dute. Beste kontu bat da, ordea, orain arte suposatu bezala egileak jarri dituen komak zuzenak izatea; gainera, egileak jarri dituen komak zuzenak izanik ere, koma konbinazio zuzen posible bat baino gehiago izan daitezke esaldiko; alegia, sailkatzaileak jarri dituen komak, jatorrizkoekin bat etorri ez arren, zuzenak izan daitezke, eta alderantziz: sailkatzaileak jarritako komak, egileak jarritakoekin bat etorriagatik, okerrak izan daitezke.

Hau guztia dela eta, ebaluazio kualitatibo osoago bat egitea erabaki genuen. Egiazki, bi ebaluazio mota egin genituen: bata tokenka, eta bestea esaldika; alegia, lehenengo ebaluazioan, token bakoitzaren ondoren koma

zihoan ala ez begiratu genuen; bigarrenean, berriz, esaldiko koma guztiak ondo zeuden ala ez begiratu genuen (esaldi baten baitan koma bat ondo jartzeak baina hurrengoa gaizki jartzeak, esaldiaren zentzua erabat alda dezakeelakoan, Shieber eta Tao (2003) lanean ikusi dugun moduan).

Ebaluazio kualitatibo honetan, hiru aukera ematen ziren ontzat: corpus originalekoa eta bi hizkuntzalarietako bakoitzarena. Hala, sailkatzaileak jarritako komak hiru aukera horiekin konparatzen ziren, tokenka lehendabizi, esaldi osoa kontuan harturik gero.

Tokenka egindako ebaluazioan (ikus 13 taula), sailkatzaileak jarritako koma bakoitza aztertzen zen, alde batetik: koma hori bi hizkuntzalarietako batek jarri bazuen edo corpuseko jatorrizkoan baldin bazetorren, ontzat ematen zen; bestalde, hiru aukeretan errepikatzen zen koma bakoitza beharrezkotzat jotzen zen, eta sailkatzaileak jarri ez bazuen, okertzat ematen zen. Irizpide hauen arabera, sailkatzaileak jarritako bost kometatik, lau ondo jartzen ditu (doitasuna = % 83,01).

13. taula. Sailkatzailearen bateragarritasuna test corpusarekiko, eta tokenka egindako ebaluazio kualitatiboaren emaitzak.

	1		
	Doit.	**Est.**	F_1
Ikask. autom. *KPI-komagabearekin* + CG erregelak	77,6	52,7	62,8
Ebaluazio kualitatiboa, tokenka	83,01	58,46	**68,61**

Esaldika egindako ebaluazioan, esaldi bat zuzentzat jotzen zen, baldin eta esaldiko koma guztiak hiru aukeretako batekin beren osoan bat baldin bazetozen, alegia, ez bazuen komarik, ez soberan, ez faltan (aukera baten eta bestearen komak nahastu gabe). Irizpide hauen arabera, 380 esaldietatik, 219 esaldi etiketatu ziren koma zuzen guztiekin (% 57,63).

3.3.7. *Erroreen analisia*

Ebaluazio kualitatiboa osatzeko, aurreko atalean ebaluatutako sailkatzailearen portaera aztertu genuen, test corpuseko esaldietan sailkatzaileak egindako iragarpenak behatuz. Jarraian, adibide argigarri batzuk aztertuko ditugu.

«&KOMA» etiketak adierazten du sailkatzaileak koma jartzea erabaki eta asmatu egin duela; «&FALTAN» etiketak, berriz, aditzera ematen du sailkatzaileak ez duela toki horretan komarik jarri, baina koma behar zuela,

betiere hiru erreferentzien arabera; azkenik, «&SOBRAN» etiketak esan nahi du sailkatzaileak koma jarri duela, komarik behar ez zen tokian.

> Adibidea 3.2
>
> Sailkatzaileak komaren bat soberan jarritako edo faltan utzitako esaldi batzuk:
>
> 1. Gurean igandeko egunkariek aste osoa ematen dute komunetik&SOBRAN bueltaka eta jiraka&FALTAN orain toalleroan&FALTAN orain erradiadorean.
> 2. Pasa ziren egiazko ospakizunak eta itxurazkoak&FALTAN etorri ziren lehen adierazpenak eta hasierako azterketak&KOMA argazkiak eta gezur ezkutatuak.
> 3. Batzuen ustez&KOMA inora ez doana&KOMA ezer egiten ez duena&KOMA eta beste batzuentzat&FALTAN gehiegi egiten duena&KOMA urrutiegi eta arinegi doana.
> 4. Haien izenak esan ondoren&FALTAN batzordea osatuta geratu zen&KOMA eta ondoren&KOMA alderdietako ordezkariek hartu zuten hitza&KOMA Aitor Gabilondo EA-EAJko alkatea lehenengoa izanik.
> 5. Sobieten Iraultzarekin hasi eta Berlingo Harresiaren birrinketaz bukatu zuen mendetik denboraren abiadura gero eta azkarragoa denez&KOMA epealdi laburrak sekulako garrantzia hartzen omen du &FALTAN Historiaren aro luzeetan orain arte suertatu ohi diren eraldaketa mantsoak gaindituz...
> 6. UEUk&FALTAN EIREk&FALTAN Euskal Adarrak&KOMA Barrutiak eta Uniekimenak «eztabaidaren erdigunean» jarri nahi dute aldarrikapena.

Koma mota ezberdinak zuzen jartzeko gai da sailkatzailea; aitzitik, 3.2 adibideko esaldietan, akatsen bat egin du sailkatzaileak.

Lehen esaldian, «*Gurean*» mintzagaiaren ondoren koma bat jartzea zilegi litzatekeen arren, ez jartzea ere ontzat eman da ebaluazio kualitatiboan. «*Komunetik*» hitzaren ondoren, sailkatzaileak jarritako koma soberan dago (baina hau ere zalantzazkoa dela esan genezake, «*bueltaka eta jiraka*» tarteki gisa uler baitaiteke). Falta direla markatutako bi komak, ordea, jarri beharrekoak dira. Hau konpontzeko, egin liteke beste CG erregela bat, «orain X, orain Y» egitura kontuan hartzen duena.

Bigarren esaldian, esaldiko elementuen ordena ez hain ohikoak eragiten du falta den komaren akatsa, gure ustez. Komaren ordez ere, jar liteke puntuazio marka gogorragoren bat (puntu eta koma, adibidez). Bestalde, informazio semantikorik gabe oso zaila deritzogu falta den koma hori detektatzeari; izan ere, esaldia zentzu honetan ere har liteke: «*Pasa ziren*

egiazko ospakizunak, eta itxurazkoak etorri ziren...». Honez gain, azpimarratu beharra dago esaldi bukaerako enumerazioari dagokion koma ondo jartzen duela sailkatzaileak.

Esaldika egindako ebaluazioa zalantzan jartzeko erakutsi ditugu hirugarren eta laugarren esaldiak. Esaldikako ebaluazioan, txartzat joko lirateke esaldi hauek, akats bat dutelako. Batzuetan, koma bakar bateko akatsak esaldi guztia desitxura dezake, baina besteetan (adibide hauetan legez), asmatutakoak baliagarriak izan daitezke eta, beraz, ebaluazioan modu positiboan adierazita agertu beharko lirateke. Beraz, uste dugu komen ebaluazio on bat osatzeko bi ebaluazioak egin beharko liratekeela, hau da esaldi mailako ebaluazioa eta token mailako ebaluazioa.

Bosgarren esaldian koma bat zuzen markatu du sailkatzaileak, eta beste bat falta zaiola ebatzi du ebaluatzaileak. Ezinezkoa iruditzen zaigu falta zaion hau, berriz ere, semantikaren ezagutzarik gabe detektatzea, zentzu handirik ez duen baina sintaktikoki zuzena den modu honetan har baitaiteke esaldia: «*epealdi laburrak sekulako garrantzia hartzen omen du Historiaren aro luzeetan, orain arte suertatu ohi diren eraldaketa mantsoak gaindituz...*».

Bukatzeko, enumerazioko kometan gertatzen dena ikusteko ekarri dugu seigarren esaldia. Enumerazioko azken hirugarren elementua komaz markatzen du sailkatzaileak, baina ez da gai enumerazioko gainerakoak (azken hirugarrenaren aurrekoak) komaz markatzeko. Test corpusean gehiagotan gertatu den fenomeno honek joera bat erakusten digu: badirudi hiru osagaiko enumerazioak ezagutu eta komaz mugatzeko gai dela sailkatzailea, baina ezdeusa dela enumerazio handiagoetan. Ikasketan erabilitako leihoan egon liteke honen arrazoia: gogora dezagun (–5,+2) leihoa erabili dugula eta horrek token bakoitzaren ondorengo bi tokenen informazioa soilik kontuan hartzea dakarrela. Leihoan tokenaren ondorengo hiru edo lau tokenen informazioa hartzeak arazo hau konpon lezakeela uste badugu ere, beste batzuk sor ditzakeela ere pentsatzekoa da.

Bukatzeko, hizkuntzalariek, etiketatze lan honen ondoren, azpimarratu nahi izan dizkiguten bi kontu ekarri nahi genituzke:

1. Esaldi batzuk anbiguoak dira, eta ez da erraza komak non doazen jakitea.
2. Batzuetan, komak gabe, puntuak edo puntu eta komak erabili nahi izan dituzte etiketatzean.

Ataza oso zaila bilakatzen da besteak beste, esaldiko elementuen ordenamendu eskasak —edo euskara batuaren ordenamendu normalenetik urruntzeak—, esaldiaren konplexutasunak, anbiguotasun semantikoak eta gainerako puntuazio ikurren erabilera ez beti zuzenak eraginda.

4. **ONDORIOAK ETA ETORKIZUNEKO LANAK**

Lan honetan euskararako koma-zuzentzaile automatiko baten lehen prototipoa sortu dugu, ikasketa automatikoko teknikak erregeletan oinarritutakoekin uztartuz.

Ikusirik hizkuntzalariek ere komak zuzen jartzeko dituzten arazoak, dela esaldi bat anbiguoa delako, dela komen konbinazio zuzen bat baino gehiago egon daitekeelako, lortu ditugun emaitzak onak direla esan dezakegu.

Era berean, koma-zuzentzaile bat sortzeko sintagmaeta perpaus identifikatzaile automatikoen beharra azaleratu dugu, eta baita garatu ere *FR Perceptron* algoritmoa erabiliz. Tresna hauek baliagarriak dira euskararen prozesamenduaren barruan egiten diren beste hainbat atazatarako, itzulpen automatikoa kasu.

Etorkizunean, aurreikusten dugu koma-zuzentzailea hobetzen saiatzeko hainbat proba egitea. Hasteko, sintagmaeta perpaus identifikatzaileak hobetzen saia gintezke, uste baitugu hauek hobetuta, koma-zuzentzailea ere hobetuko litzatekeela. Bestalde, koma bat jarri ala ez erabakitzeko, erabakigarria izan daiteke aurreko komen informazioa kontuan hartzea ere. HMM edo CRF algoritmoek, dituzten berezko ezaugarriengatik, egokiak dirudite modu honetako atazak ebazteko.

Halaber, komaren emaitza klaseen arteko desoreka konpontzeko marjina aldagarriak erabil litezke; hau da, bi klaseen arteko marjina handieneko hiperplanoa bilatu ordez, klase bakoitzerako marjina bana bila daitezke, klase bateko adibideen kopuru handiagoak emandako *abantaila* nolabait orekatzeko (Li *et al*., 2009, 2002).

Azkenik, komak zuzentzeaz gain, aurreikusten dugu komak jartzen ikasten laguntzeko modulua garatzea ere aurreikusten dugu. Ezinbestekoa litzateke horretarako, koma non jarri behar den jakiteaz gain, ezagutzea zein den koma leku horretan jartzeko arrazoia. Hori dela eta, koma bakoitzaren sintaxi zeregina finkatu beharko litzateke, (Delden eta Gomez, 2002) eta (Srikumar *et al*., 2008) lanen ildotik.

5. **ESKER ONAK**

Espainiako Heziketa eta Zientzia ministerioak diruz lagundutako lana (TIN2009-14675-C03-01). Euskal Herriko Unibertsitateko SGI/IZO-SGIker atala (Europako fondo sozialeko garapen eta berrikuntza sailak, MCyTek eta Eusko Jaurlaritzak diruz lagundua) ere eskertu nahi genuke, baliabide konputazionalak uzteko izan duen eskuzabaltasunarengatik.

6. **BIBLIOGRAFIA**

ABNEY S. 1997. «Part-of-speech tagging and partial parsing». *Corpus-Based Methods in Language and Speech Processing. ELSNET.*

ADURIZ I., ARANZABE M., ARRIOLA J.M., DIAZ DE ILARRAZA A., GOJENOLA K., ORONOZ M., eta URIA L. 2004. «A cascaded syntactic analyser for Basque». In Gelbukh A., editor, *Computational Linguistics and Intelligent Text Processing: 5th International Conference CICLing2004, Seoul, Korea, February 15-21*, 2945 lib. of *Lecture Notes in Computer Science*, 124-134. Springer-Verlag GmbH.

ADURIZ I., ARRIETA B., ARRIOLA J.M., DIAZ DE ILARRAZA A., IZAGIRRE E., eta ONDARRA A. 2006. *Muga gramatikaren optimizazioa*. Barnetxostena UPV/EHU / LSI / TR 9-2006, University of the Basque Country, Informatika Fakultatea, Donostia.

ADURIZ I. eta DIAZ DE ILARRAZA A. 2003. «Morphosyntactic disambiguation and shallow parsing in computational processing of Basque». *Inquiries into the lexicon-syntax relations in Basque*, 1-21.

AGIRRE E., ALEGRIA I., ARREGI X., ARTOLA X., DIAZ DE ILARRAZA A., URKIA M., MARITXALAR M., eta SARASOLA K. 1992. «XUXEN: A spelling checker/corrector for Basque based on two-level morphology». *Proceedings of ANLP'92*, 119-125, Povo Trento.

ALDEZABAL I., ARANZABE M., ATUTXA A., GOJENOLA K., SARASOLA K., eta ZABALA I. 2003- *Hitz-hurrenkeraren azterketa masiboa corpusean. Barnetxostena*, EHU.

ALDEZABAL I. 2004. *Aditz-azpikategorizazioaren azterketa sintaxi partzialetik sintaxi osorako bidean. 100 aditzen azterketa, Levin-en (1993) lana oinarria hartuta, eta metodo automatikoak baliatuz.* Doktoretza-tesia, Euskal Filologia Saila, Euskal Herriko Unibertsitatea, Leioa.

ALEGRIA I. 1995. *Euskal morfologiaren tratamendu automatikorako tresnak.* Doktoretza-tesia, Informatika Fakultatea. UPV-EHU, uztaila 1995.

ANDO R.K. eta ZHANG T. 2005. «A high-performance semi-supervised learning method for text chunking». *ACL '05: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, 1-9, Morristown, NJ, USA. Association for Computational Linguistics.

ARANZABE M.J. 2008. *Dependentzia-ereduan oinarritutako baliabide sintaktikoak: zuhaitz-bankua eta gramatika konputazionala.* Doktoretza-tesia, Euskal Filologia Saila, Euskal Herriko Unibertsitatea.

ARRIETA B. 2010. *Azaleko sintaxiaren tratamendua ikasketa automatikoko tekniken bidez: euskarako sintagmeneta perpausen identifikazioa eta bere erabilera koma-zuzentzaile batean.* Doktoretza-tesia, Lengoaia eta Sistema Informatikoak Saila, Euskal Herriko Unibertsitatea.

BALDWIN T. eta JOSEPH M. 2009. «Restoring punctuation and casing in english text». *AI '09: Proceedings of the 22nd Australasian Joint Conference on Advances in Artificial Intelligence*, 547-556, Berlin, Heidelberg, Springer-Verlag.

BANKO M. eta BRILL E. 2001. «Scaling to very very large corpora for natural language disambiguation». *ACL '01: Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, 26-33, Morristown, NJ, USA, Association for Computational Linguistics.

BAYRAKTAR M., SAY B., eta AKMAN V. 1998. «An analysis of English punctuation: the special case of comma». *International Journal of Corpus Linguistics*, **3**(1):33-57.

BLACK E., ABNEY S., FLICKENGER D., GDANIEC C., GRISHAM R., HARRISON P., HINDLE D., INGRIA R., JELINEK F., KLAVANS J., LIBERMAN M., MARCUS M., ROUKOS S., SANTORINI B., eta STRZALKOWSKI T. 1991. «A procedure for quantitatively comparing the syntactic coverage of English grammars». *Proceedings of DARPA Workshop on Speech and Natural Language*.

BRILL E. 1995. «Transformation-based error-driven learning and Natural Language Processing: A case study in part of speech tagging». *Computational Linguistics*, **21**(4):543-565.

CARLETTA J. 1996. «Assessing agreement on classication tasks: the kappa statistic». *Computational Linguistics*, **22**(2).

CARRERAS X. 2005. *Learning and Inference in Phrase Recognition: A Filtering-Ranking Architecture using Perceptron*. Doktoretza-tesia, Polytechnic University of Catalunya.

CARRERAS X., MÁRQUEZ L., eta CASTRO J. 2005. «Filtering-ranking perceptron learning for partial parsing». *Machine Learning Journal, Special Issue on Learning in Speech and Language Technologies*, **60**(1-3):41-71.

CARRERAS X. eta MÁRQUEZ L. 2003. «Phrase recognition by filtering and ranking with perceptrons». *Proceedings of the International Conference on Recent Advances in Natural Language Processing, RANLP*. Borovets, Bulgaria.

DELDEN S.V. eta GOMEZ F. 2002. «Combining finite state automata and a greedy learning algorithm to determine the syntactic roles of commas». *Proceedings of the 14th IEEE International Conference on Tools with Artificial Intelligence*, Washington D.C. USA.

DIETTERICH T.G. 1998. *Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms*, 1895-1924. Number 10 in 7. MIT press journals.

ERDOZIA K., LAKA I., MESTRES-MISSE A., eta RODRIGUEZ-FORNELLS A. 2009. «Syntactic complexity and ambiguity resolution in a free word order language: behavioral and electrophysiological evidences from basque». *Brain and Language*, 1-17.

EVERITT B. 1992. *The analysis of contingency tables*. Chapman and Hall.

EZEIZA N. 2002. *Corpusak ustiatzeko tresna linguistikoak. Euskararen etiketatzaile sintaktiko sendo eta malgua*. Doktoretza-tesia, University of the Basque Country, Donostia.

GARZIA J. 1997. *Joskera lantegi*. Euskal Autonomia Erkidegoko Administrazioa, IVAP.

GOJENOLA K. 2000. *Euskararen sintaxi konputazionalerantz. Oinarrizko baliabideak eta beren aplikazioa aditzen azpikategorizazio-informazioaren erauzketan eta erroreen tratamenduan*. Doktoretza-tesia, Informatika Fakultatea. Euskal Herriko Unibertsitatea, Donostia.

GUINOVART F.J.G. 1996. *Fundamentos y límites de los sistemas de verificación automática de la sintaxis y el estilo*. Doktoretza-tesia, Universidade de Santiago de Compostela.

HARDT D. 2001. «Comma checking in Danish». *Corpus Linguistics*, Lancaster (England).

HIDALGO B. 1994. *Hitzen ordena euskaraz*. Doktoretza-tesia, Euskal Herriko Unibertsitatea.

JAKUBICEK M. eta HORAK A. 2010. «Punctuation detection with full syntactic parsing». *Proceedings of CICLing-2010. 11th International Conference on Intelligent Text Processing and Computational Linguistics*, Romania.

JOACHIMS T. 1998. «Text categorization with support vector machines: learning with many relevant features». In N'edellec C. eta Rouveirol C., editors, *Proceedings of ECML-98, 10th European Conference on Machine Learning*. Springer.

KARLSSON F., VOUTILAINEN A., HEIKKILA J., eta ANTTILA A. 1995. *Constraint Grammar: Language-independent System for Parsing Unrestricted Text*. Prentice-Hall, Berlin.

LEE Y. eta WU Y. 2007. «A robust multilingual portable phrase chunking system». *Expert Syst. Appl.*, **33**(3).

LI Y., BONTCHEVA K., eta CUNNINGHAM H. 2009. «Adapting svm for data sparseness and imbalance: a case study in information extraction». *Natural language Engineering*.

LI Y., ZARAGOZA H., HERBRICH R., SHAWE-TAYLOR J., eta KANDOLA J. 2002. «The perceptron algorithm with uneven margins». *ICML02: proceedings of the 19th international conference on Machine Learning*, San Francisco, USA, Morgan Kaufmann Publishers Inc.

MARCUS M., MARCINKIEWICZ M.A., eta SANTORINI B. 1993. «Building a large annotated corpus of English: The Penn treebank». *Computational Linguistics*, **19**(2).

MAYOR A., ALEGRIA I., DE ILARRAZA A.D., LABAKA G., LERSUNDI M., eta SARASOLA K. 2009. «Evaluación de un sistema de traducción automática basado en reglas o por qué bleu sólo sirve para lo que sirve». *Revista de la Asociación Española para el Procesamiento del Lenguaje Natural*.

MILENOVA B., YARMUS J., eta CAMPOS M. 2005. «Svm in oracle database 10g: Removing the barriers to widespread adoption of support vector machines». *Proceeding of the 31st VLDB Conference*, Trondheim, Norway.

MOLINA A. 2003. *Desambiguación en procesamiento del lenguaje natural mediante técnicas de aprendizaje automático*. Doktoretza-tesia, Universidad Politécnica de Valencia, Valencia.

MOTKHTAR S.A., CHANOD J., eta ROUX C. 2002. «Robustness beyond shallowness: incremental deep parsing». *Natural Language Engineering*, **8**(2-3):121-144.

NGUYEN V.V., NGUYEN M.L., eta SHIMAZU A. 2009. «Clause splitting with conditional random fields». *Information and Media Technologies*, **4**(1): 57-75.

NUNBERG G. 1990. *The linguistics of Punctuation*. Center for the study of language information (CSLI), Lecture notes: no. 18, University of Chicago Press,.

ODRIOZOLA, J.C. 2005. *Puntuazioa gramatikagai (La puntuación como objeto de estudio de la gramática)*. Nerekin jaio nun. Txillardegiri omenaldia (Homenaje a Txillardegi) Iker (17): 353-378. Bilbo.

ODRIOZOLA J.C. eta ZABALA I. 1993. *Hitz-ordena, galdegaia eta komaren erabilera*. Euskal Herriko Unibertsitateko Argitarapen Zerbitzua, Bilbo.

ORONOZ M. 2009. *Euskarazko errore sintaktikoak detektatzeko eta zuzentzeko baliabideen garapena: datak, postposizio-lokuzioak eta komunztadura*. Doktoretza-tesia, Lengoaia eta Sistema Informatikoak Saila, Euskal Herriko Unibertsitatea.

RAM R.V.S. eta DEVI S.L. 2008. «Clause boundary identification using conditional random fields». *Computational Linguistics and Intelligent Text Processing*, **4919**:140-150.

REHUREK R. eta SOJKA P. 2010. «Automated classification and categorization of mathematical knowledge». *Intelligent Computer Mathematics*.

SANG E.T.K. eta BUCHHOLZ S. 2000. «Introduction to the conll-2000 shared task: Chunking». *Proceedings of Computational Natural Language Learning*, Lisbon (Portugal).

SANG E.T.K. eta DEJEAN H. 2001. «Introduction to the conll-2001 shared task: Clause identification». *Proceedings of Computational Natural Language Learning*, Toulouse (France).

SAY B. eta AKMAN V. 1996. «Current approacehs to punctuation in Computational Linguistics». *Computers and the Humanities*, **30**(6):457-469.

SHI Y. eta WANG M. 2007. «A dual-layer crfs based joint decoding method for cascaded segmentation and labeling tasks». *IJCAI'07: Proceedings of the 20th international joint conference on Artifical intelligence*, 1707-1712, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

SHIEBER S.M. eta TAO X. 2003. «Comma restoration using constituency information». *Proceedings of HLT-NAACL*.

SRIKUMAR V., REICHART R., SAMMONS M., RAPPOPORT A., eta ROTH D. 2008. «Extraction of entailed semantic relations through syntax-based comma resolution». *Proceedings of ACL-08: HLT*, 1030-1038, Columbus, Ohio, June 2008. Association for Computational Linguistics.

TANEV H. eta MITKOV R. 2002. «Shallow language processing architecture for bulgarian». *Proceedings of the 19th international conference on Computational linguistics*, 1-7, Morristown, NJ, USA. Association for Computational Linguistics.

TAPANAINEN P. 1996. *The Constraint Grammar parser CG-2*. Publications of the University of Helsinki, 27, Helsinki.

URIA L. 2009. *Euskarazko erroreen eta desbideratzeen analisirako lan-ingurunea. Determintzaile-erroreen azterketa eta prozesamendua*. Doktoretza-tesia, Euskal Filologia Saila, Euskal Herriko Unibertsitatea.

URKIA M. 1997. *Euskal morfologiaren tratamendu informatikorantz*. Doktoretza tesia, Filologia eta Historia-Geografia Fakultatea. UPV-EHU, uztaila 1997.