eman ta zabal zazu
Universidad del País Vasco
Euskal Herriko Unibertsitatea
INFORMATIKA FAKULTATEA
FACULTAD DE INFORMÁTICA

Informatika Ingeniaritzako Gradua

Konputazioa

Gradu Amaierako Lana

Kontzeptuen arteko erlazio-erauzle automatiko baten inplementazioa

Egilea

Oscar Sainz Jimenez

2019

eman ta zabal zazu
Universidad del País Vasco
Euskal Herriko Unibertsitatea
INFORMATIKA FAKULTATEA
FACULTAD DE INFORMÁTICA

Informatika Ingeniaritzako Gradua

Konputazioa

Gradu Amaierako Lana

Kontzeptuen arteko erlazio-erauzle automatiko baten inplementazioa

Egilea

Oscar Sainz Jimenez

Zuzendariak

Itziar Aldabe eta Montserrat Maritxalar

Laburpena

Testu baten gaineko bi termino izanik, beraien artean ze erlazio-mota dagoen, baldin badago, adieraziko duen sistema baten garapena eraman dugu aurrera. Horretarako, ikasketa automatikoan eta hizkuntzaren prozesamenduan oinarritutako teknikak aztertu eta erabili dira. Sistemaren helburua edo motibazioa galderen sorkuntzarako beste sistema handiago baten parte izatea da, galdera esanguratsuagoak sortzeko asmoz.

Alde batetik, automatikoki etiketatutako corpus bat sortu dugu, biologia domeinukoa. Corpusa sortzeko urruneko gainbegiraketan oinarritu gara, ConceptNet ezagutza-basea eta Wikipediako artikuluak erabiliz. Automatikoki sortutako corpusa zaratatsua denez, hainbat iragazketa aplikatu dizkiogu zarata hori desagerrarazteko asmoarekin.

Bestetik, erlazio-erauzle sistema bat garatu dugu ikasketa-automatikoko tekniketan oinarrituta. Erabilitako sistema algoritmo tradizionaletan oinarritzen denez, ezaugarri-ingeniaritza fase bat egon da. Behin ezaugarriak definituta, sailkatzailearen aukeraketa-prozesu bat eman dugu. Hasiera-lerro bat definitu dugu, eta, iteratiboki, sistema horren gainean hobekuntzak aplikatu ditugu.

Azkenik, sistema ebaluatu dugu modu kuantitatibo batean. Laburbilduz, eta zerotik garatu den sistema bat dela jakinda, lortutako emaitzak nahiko onak izan direla ondorioztatu dezakegu. Hala ere, sistemaren akatsak eta hobetu daitezkeen gauzak azaltzen ditugu errore-analisi batean.

Gaien aurkibidea

Irudien aurkibidea

Taulen aurkibidea

1. KAPITULUA

Sarrera

Hezkuntza arloan, testu bat ulertu ahal izateko irakurketa ez da nahikotzat ematen: testuaren gaineko galderak erantzutea ezinbestekoa dela esan ohi da testua sakon ulertzeko. Ulermen-prozesu batean ikasle bati eskaintzen zaizkion galderek zeresan handia dute ikaslearen ulermen-prozesuan. Galdera "onak" sortzea, ordea, erronka handia izaten da materiala prestatzen duten adituentzat. Hortaz, erronka handiagoa da galderak automatikoki sortu nahi baditugu. Mostow and Chen (2009)-ek eta M. Olney et al. (2012)-ek oinarri pedagogikoa duten galderak automatikoki sortzeko eredua aurkezten dute. Horretarako, testu batetik erauzitako ezagutza entitateen arteko erlazioen bidez errepresentatzen dute, beti ere testuaren ikuspegi orokor batetik.

Testu baten gainean galdera onak egin ahal izateko, aldez aurretik, testuaren ideiak, gertaerak, entitateak, pertsonaiak, eta abar luze bat identifikatzea garrantzitsua izaten da. Horretarako, ezinbestekoa da hizkuntzaren prozesamendurako teknologiaren garapena. Testuetako ezagutzaren detekzioa eta errepresentazioa era egituratu batean funtsezkoa suertatzen da, testuaren ulermenerako galdera baliagarriak sortu nahi baditugu. Hori dela eta, proiektu honen azkeneko helburua testuaren ulermenerako giltzarri den informazioaren erauzketarako tresnak garatzea da.

Beraz, proiektu hau hizkuntzaren prozesamenduan kokatzen da, idatzizko dokumentuetan agertzen den informazioa erauzteko teknologiaren garapen arloan zehazki. Konkretuki, testu batean irakurketa-prozesuan zehar ulermenerako baliagarri den informazioa detektatzea du helburu lan honek.

Bi helburu nagusi planteatzen ditugu proiektu honetan: alde batetik, ikasketa automati-

koan oinarritutako informazio-erauzle baten garapena, konkretuki, bi entitateen arteko erlazioa erauziko duen sistema bat; eta, bestetik, aipatutako sistema entrenatzeko erabiliko dugun corpus baten etiketazioa. Aipatzeko, domeinu eta hizkuntza zehatz batera mugatu dugula proiektua, hain zuzen ere, biologiako domeinura eta ingeles hizkuntzara.

Aipatutako corpusa sortzeko arrazoia gure ataza eta domeinurako corpus publikorik existitzen ez delako da. Hala ere, corpus bat etiketatzea garestia da, eskuzko lan bat dakarrelako. Hori dela eta, Mintz et al. (2009)-en lanean aipatzen duten urruneko gainbegiraketan (2.1. atalean azaldua) oinarritu gara corpusa etiketatzeko. Ideia hori inplementatzeko prozesua bi baliabidez hornitzen da: dokumentu multzo handi bat eta ezagutza-base bat. Prozesu horren inplementazioa eta emaitzak 4. kapituluan daude azalduta.

Erlazio-erauzleari dagokionez, lehenengo pausoa antzeko lanen azterketa bat izan da, bertan, proiektu honen eredu izan den Mintz et al. (2009). artikulua eta beste batzuk aztertu ditugu. Behin azterketa-prozesua burutu dugula, sistema garatu dugu. Gure erlazio-erauzle sistemaren garapenean lehendabiziko pausoa oinarri-lerro bat garatzea izan da, gero, iteratiboki horren gainean hobekuntzak inplementatzeko.

Gure sistema nolako errendimendua duen ikusteko ebaluazio-prozesu bat eraman dugu aurrera. Prozesua ebaluazio kuantitatiboan oinarritu da. Horretarako, ikasketa automatikoko ebaluazio-metrikak erabili ditugu, adibidez: doitasuna, estaldura, F1, etab. Ez bakarrik azkeneko sistema, oinarri-lerroa eta horrek jasan dituen hobekuntzak ere ebaluatu ditugu, gero, horien konparaketa bat egiteko. Azkenik, gure sistema onenaren errore-analisi bat egin dugu, sistemak zertan egiten du oker eta non egon daitezkeen hobekuntzak ikusteko.

Azkenik, proiektu honetan hainbat ekarpen sortu ditugu, alde batetik, erlazio-erauzketa atazarako biologia domeinuko ingelesezko corpus bat eta horren gainean entrenaturiko erlazio-erauzle sistema bat, bestetik, eta garrantzitsuena, prozesu osoaren garapen-lerroaren diseinua eta inplementatzeko beharrezkoa den kodea. Ekarpen guzti horiek publikoki argitaratzea da ideia, gero, beste edonork esperimentuak errepikatzeko aukera izateko. Kodea jadanik GitHub-en [1] dago argitaratua.

[1] www.github.com/osainz59/Erlazio_erauzlea

2. KAPITULUA

Artearen egoera

Atal honetan, proiektuarekin zerikusia duten atazen artearen egoera azalduko dugu. Hain zuzen ere, urruneko gainbegiraketa eta horren aplikazioa erlazio-erauzketan, sostengu bektore makinak eta ebaluazio-metrikak aipatuko ditugu.

2.1 Urruneko gainbegiraketa

Urruneko gainbegiraketa corpus bat modu automatikoan etiketatzeko erabiltzen den teknika bat da gehienbat erlazio-erauzketara zuzendua. Dokumentu multzo eta ezagutza-base batez baliatuta ezagutza-basean agertzen diren erlazioak dokumentu multzoko adibideekin parekatzean datza.

Teknika Mintz et al. (2009)-ek proposatutako hipotesian oinarritzen da: esaldi bateko bi termino ezagutza-base batean erlazionaturik badaude, hau da, existitzen bada ertz bat bi termino horien artean, orduan esaldi horrek bi terminoen arteko erlazioa adierazten du.

Entitate 1	Erlazioa	Entitate 2
Heart	At Location	Body
Azalea	Is A	Plant
Valve	Part Of	Heart
Wing	Used For	Fly

2.1 Taula: Conceptnet-eko adibide batzuk.

Ezagutza-base batean bi entitate eta beraien arteko erlazioa osatzen duten hirukoteak ager-

Esaldia
The procedure is performed by incision of a suitable vein into which the electrode lead is inserted and passed along the vein, through the **valve** of the **heart**, until positioned in the chamber.

2.1 Irudia: Wikipediako artikulu batetik erauzitako adibide bat <Valve, Heart, PartOf> hirukotea erabilita.

tzen dira. Hirukote horien adibide bat <*Valve*, *Heart*, *PartOf*> da (ikusi 2.1. taula adibide gehiagorako). Urruneko gainbegiraketaren arabera hirukotea osatzen duten entitatebikotea dokumentu multzoan bilatuz entrenamendurako balioko diren adibideak lortuko ditugu, adibidez, 2.1 irudian agertzen den esaldia.

2.1.1 Urruneko gainbegiraketaren arazoak

Urruneko gainbegiraketa aurkezten da ikasketa gainbegiratua, erdi-gainbegiratua eta ez-gainbegiratuak dituzten arazoei soluzio bat emateko (ikusi 2.1.2. atala). Hori horrela izan arren urruneko gainbegiraketak ere hainbat arazo ditu:

Adibide negatiboen sorkuntza
Adibide negatiboen sorkuntza ingenieritza lan bat baino artelan bat da. Adibidez, Mintz et al. (2009)-ek jarraitzen duten estrategia, ezagutza-base batean erlaziorik ez duten bi entitate ausaz aukeratu eta horiek negatibotzat eman.

Sistema batek ondo funtziona dezan adibide positiboez aparte negatiboak ere behar ditu; bestela, sistemak behartuta egongo lirateke beti erlazio baten aurrean daudela esatera, eta horrek ez du zertan egia izan behar.

Ezagutza-basearen desoreka
Ezagutza-base bateko desoreka esanguratsuena erlazioen arteko kopuru desberdintasuna izaten da. Erlazio batzuk ohikoagoak dira beste batzuk baino, eta, desberdintasun hori handiegia bada, erlazio horiek beste batzuk estaltzeko probabilitatea handiagoa egongo da.

Urruneko gainbegiraketa guztiz mendekoa da ezagutza-basearen edukiarekiko; horrek esan nahi du ezagutza-baseak dituen desoreka arazoak sortuko den corpusera hedatuko direla.

Adibide zaratatsuak
Ezagutza-basean esleitutako erlazioa eta esaldian agertzen den erlazioa bat ez datoze-

nean gertatzen dira, baita esaldia erlaziorik erakusten ez dutenean ere. Adibide zaratatsuak urruneko gainbegiraketak duen arazorik larriena da, eta hurrengo arrazoiengatik ager daitezke:

1. **Erlaziorik ez egotea adibidean:** bi entitateak esaldi berean agertzea baina esaldiak erlazio hori ez erakusteari deritzo, hau da, urruneko gainbegiraketaren atzetik dagoen hipotesia ez betetzea. Adibide zaratatsuak sortzeko arrazoiengandik hau izaten da nagusiena.

2. **Ezagutza-basearen osotasun-eza:** ezagutza-baseak eskuz sortutakoak dira, eta, askotan, kasu posibleen estaldura ez dago bermatuta. Demagun esaldi batean erlazionaturiko bi entitate agertzen direla, baina erlazio hori ez dela agertzen ezagutza-basean, orduan, urruneko gainbegiraketaren arabera esaldi horrek ez du erlaziorik erakusten. Ondorioz gaizki etiketaturiko adibideak sor daitezke.

3. **Erlazio anitza:** kasu honetan ezagutza-basean bi entitateen artean erlazio bat baino gehiago daudenean gertatzen da. Urruneko gainbegiraketa bakarrik erabilita ez da posible izaten erauzitako adibide bakoitzean zein erlazio agertzen den esatea.

2.1.2 Erlazio-erauzketa urruneko gainbegiraketa erabiliz

Erlazio-erauzketa hizkuntzaren prozesamenduko ataza bat da, hain zuzen ere, informazio-erauzketa arloan dago kokatuta. Bi izan dira ataza honi aurre egiteko planteamendu nagusiak: gainbegiratua eta erdi-gainbegiratua (*supervised* eta *semi-supervised* ingelesez hurrenez-hurren). Bi planteamendu horiek muga esanguratsuak aurkezten dituzte. Gainbegiratuen kasuan etiketaturiko corpus baten beharra dago, corpus horren lorpena oso garestia izan daiteke eskuzko lana dakarrelako. Muga horrek erlazio-erauzketako sistema zabaltzea zailtzen du, erlazio berriak sartzeko momentuan etiketaturiko datu-multzo berriak behar direlako. Erdi-gainbegiratuen kasuan, txikia bada ere, hasierako etiketaturiko corpus baten beharra dago.

Kontrajarrian metodo ez-gainbegiratuak (*unsupervised* ingelesez) daude. Aipatutako metodoen kasuan eskuzko lanik egitea ez da beharrezkoa, baina lortzen dituzten emaitzak interpretatzeko zailak izaten dira, baita erlazio multzo, eskema edo ontologia batera hedatzea ere.

Hori dela eta, Mintz et al. (2009) artikuluan aurkezten da urruneko gainbegiraketa (*distant supervision* ingelesez) paradigma hau. Aurretik azaldu dugun bezala, ezagutza-base

eta dokumentu multzo handi batez baliatuz automatikoki etiketaturiko corpus baten sorkuntzarako prozesua definitzen dute artikuluan. Modu horretan beste paradigmek duten eskuzko corpus etiketaturiko baten beharra guztiz desagerrarazten da.

2.1.1 atalean aipatutako arazoei aurre egiteko, hurrengo paragrafoetan Smirnova and Cudré-Mauroux (2018) lanean jasotzen diren hobekuntzak aipatuko ditugu.

Hobekuntzak hiru mota edo ataletan sailka daitezke: zarata-iragazketan oinarritutako hobekuntzak, sare neuronal sakonetan oinarritutakoak eta informazio lagungarrian oinarritutakoak.

Zarata-iragazketan oinarritutako hobekuntzek entrenamendurako erabiliko den corpusa duen zaratan jartzen dute arreta. Batez ere, hiru nabarmentzen dira besteen gainetik:

- **Gutxienez bat** (*At-least-one* ingelesez) modeloak. Modelo hauek urruneko gainbegiraketaren hipotesiaren aldaera batean oinarritzen dira: erlaziodun entitate bikote baten dokumentu multzoko adibideen arteko adibide batek gutxienez erlazioa erakusten du. Hori dela eta, sailkapena bi prozesutan banatzen da, alde batetik, lortutako adibideak erlaziorik erakusten duten edo ez sailkatzea, eta, bestetik, entitate parea zein erlazio erakusten duen sailkatzea.

- **Gai-modelaketan** (*Topic modeling* ingelesez) oinarritutako modeloak. Modelo hauek gai-modelaketa atazean ohikoak diren eredu probabilistikoak erabiltzen dituzte. Orokorrean Latent Dirichlet Allocation (LDA) ereduan oinarritzen dira (ikusi 4.1.2 atala). Metodo hauek oso erabiliak izan dira paradigma ez-gainbegiratuan Yao et al. (2011)-ek erakusten duten bezala. Hala ere, bertan erakusten dute lortutako informazioa urruneko gainbegiraketan baliagarria dela adibide zaratatsuak eta onak bereizteko.

- **Patroi-korrelazioan** (*Pattern Correlations* ingelesez) oinarritutakoak. Metodo hauen ideia da, erlazio bakoitzarentzat ezagunak diren patroi zaratatsuen zerrenda bat edukita, patroi zaratatsuak jarraitzen dituzten adibideak ezabatzea. Takamatsu et al. (2012)-ek adibidez, eredu probabilistiko baten bitartez erauzten dituzte patroi zaratatsuak.

Sare neuronal sakonetan oinarritutako metodoak ikasketa automatikoko algoritmoak ordezkatzen hasi dira ia ataza guztietan, emaitzak modu nabarmen batean hobetuz. Sare neuronal sakonen artean sare konboluzionalak daude, sare horiek oso erabiliak izan dira batez ere irudien sailkapen atazetan. Irudien gainean kurbak eta ertzak antzemateko gai

dira, gero, elementu konplexuago batzuk identifikatu ahal izateko. Testuan berriz *n-gram*-ak bezalako ezaugarriak erauzteko ahalmena dute.

Zeng et al. (2015)-ek aurkezten dute Zatikatutako Neurona Sare Konboluzionala (*Piecewise Convolutional Neural Network* edo PCNN) erlazio-erauzketa atazarako. Bertan erakusten dute sare neuronal sakonen ahalmena, hain zuzen ere, sareak ikasitako ezaugarriak pertsonak eskuz diseinatutakoak baino hobeak direla.

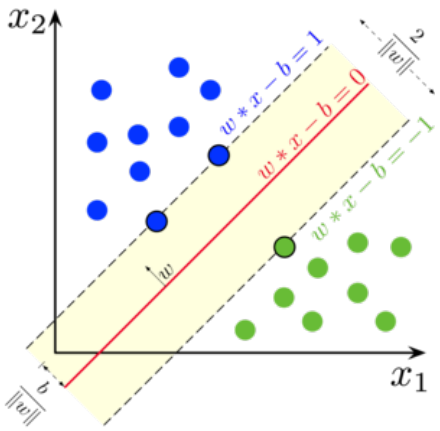
Informazio lagungarrian oinarritutako metodoak, corpusetik kanpo dagoen informazioa erabiltzen dute. Informazio lagungarri hori mota desberdinetakoa izan daiteke, adibidez:

- **Eskuz etiketaturiko adibideak** erabiltzea automatikoki etiketatutakoekin batera. Ideia horri gainbegiraketa zuzena *Direct Supervision* ingelesez) deitu diote. Angeli et al. (2014)-ek erakusten dute kopurua txikia izan arren eskuz etiketaturiko adibide batzuk entrenamendura gehituta emaitzak nahiko hobetzen direla. Ideia horrekin erlazionatuta dagoen beste bat Pershina et al. (2014) aurkezten dute, gidatutako urruneko gainbegiraketa (*Guided DS* ingelesez) hain zuzen ere. Beraien planteamendua da eskuz etiketaturiko adibideak automatikoki etiketatuak baino askoz esanguratsuak dira. Hori dela eta, eskuz etiketaturiko adibideekin bakarrik, ezaugarri-aukeraketa (*feature selection* ingelesez) moduko bat inplementatzen dute, gero, entrenamenduan gida-lerro bezala erabiltzeko.

- **Entitate-izenen identifikazioa** (*Named Entity Recognition* edo NER ingelesez) erabiltzea. Entitate-izenen identifikazioaren helburua da testu baten gainean ezagunak diren entitateak aurkitzea; entitate hauek laburtuta edo siglen bidez ere adieraz daitezke. Urruneko gainbegiraketa aplikatzeko testuko entitateak normalizatuta egotea beharrezkoa da ezagutza-basearekin erlazionatu ahal izateko, horregatik entitate horien identifikazio egokia da hain garrantzitsua. Testu bateko entitateak ezagutza-base bateko sarrerekin lotzea, entitate-lotura (*entity-linking* ingelesez) bezala ezagutzen den ataza da. Shen et al. (2015) artikuluan ataza honi buruzko hainbat arazo, teknika eta soluzio aurkezten dituzte.

2.2 Sostengu bektore makinak

Proiektuan zehar erabili dugun ikasketa automatikorako algoritmoa Sostengu Bektore Makina SBM (Support-Vector Machine ingelesez) familiako eredu linearra izan da.



2.2 Irudia: Bi klase linealki banagarrien arteko sailkapena SBM baten bitartez.

Algoritmoa ezaugarri-espazioan definitutako hiperplano batean oinarritzen da. Hiperplano hori aurkitzeko klase desberdineko gertueneko instantziak, hots, sostengu bektoreak aurkitu behar dira. Algoritmoak hiperplanoa instantzia horien artean kokatuko du, non, hiperplanoa eta instantzien arteko distantzia (*marjin*-a) handiena den (ikusi 2.2 irudia).

Aipatutako hiperplanoa zuzen bat bezala definitzen da ezaugarrien espazioan:

$$f(x) = w^T \cdot x + b \tag{2.1}$$

Sailkatzaileak ikasi beharreko parametroak honako hauek dira: $\vec{w}$ ezaugarri bakoitzari emango zaion pisua eta b jatorrizko puntuarekiko distantzia. Sailkapen bitarra egiteko hurrengo funtzioa erabiltzen da, bertan 2.1 funtzioa erabilita:

$$\bar{y}(x) = \begin{cases} +1 & \text{baldin eta } f(x) \geq 0 \\ -1 & \text{bestela} \end{cases} \tag{2.2}$$

Klase anitzeko sailkapena

SBM algoritmoak sailkapen bitarra egiteko daude pentsatuta, horregatik, erronkaren planteamendua aldatu beharra dago. Kasu hauetan erabiltzen den paradigma *bat besteen aurka* (*One Versus Rest* ingelesez) deitzen da. Paradigma horren arabera, klase bakoitzarentzat sailkatzaile bat entrenatzen da non adibide positiboak klase horretako adibideak diren eta negatiboak beste guztiak, beraz, n klase desberdin dituen sailkapen batean $f_1, \ldots, f_n$ hiperplano ikasi beharko dira.

Klase anitzeko sailkapena egiteko erabiliko dugun funtzioa:

$$\bar{y}(x) = \arg \max_{i=1,\ldots,n} f_i(x) \tag{2.3}$$

SBMen optimizazoa

Hiperplanoak ikastea-optimizazio problema bat bezala aurkezten da hurrengo helburu-funtzioarekin:

$$\min_{w} \quad \frac{1}{2} w^T w + C \sum_{i=1}^{l} \mathcal{L}(w; x_i, y_i) \tag{2.4}$$

non, $C > 0$ zigortze-parametroa den eta $y_i \in [0,1]$ i. adibidearen klasea den. 2.4 funtzioan bi zati bereiztu behar dira, alde batetik, ereduari orokortzen lagunduko dion erregularizazio-zigorra, 2.4 funtzioaren batuketaren lehenengo zatia, eta, bestetik, $\mathcal{L}$ *loss* funtzioa. Proiektuan *squared hinge* izenarekin ezagutzen den *loss* funtzioa erabiltzea erabaki dugu:

$$\mathcal{L}(w; x_i, y_i) = \max(0, 1 - y_i \cdot (w^T x_i + b))^2 \tag{2.5}$$

2.3 Ebaluazio-metrikak

Proiektuan zehar erabiliko ditugun ebaluazio-metrikak hurrengoak dira: Doitasuna (Precision), Estaldura (Recall) eta F1 neurria. Neurri horiek oso ohikoak dira ikasketa automatikoko atazetan. Aipatutako neurriak azaltzeko demagun sailkapen bitar baten aurrean gaudela non A eta B klaseak diren.

A klasearen doitasuna:

A klase bezala iragarri direnen artean benetan A klasekoak zenbat diren adierazten du. Neurri honen bitartez neur daiteke zenbaterainoko ziurtasuna duen gure sailkatzaileak.

$$doitasuna = \frac{asmatutakoak}{iragarritakoak}$$

A klasearen estaldura:

A klasekoak direnen artean A klasekoak bezala zenbat iragarri diren azaltzen du. Neurri honek gure sailkatzaileak planteatutako problemara egokitzeko duen ahalmena adierazten du.

$$estaldura = \frac{asmatutakoak}{benetakoak}$$

F1 neurria doitasuna eta estalduraren arteko batezbesteko harmonikoa da.

$$F1 = 2 * \frac{doitasuna * estaldura}{doitasuna + estaldura}$$

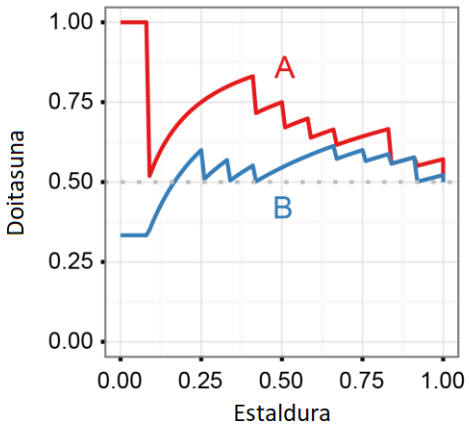
Klase anitzeko problemetan
Sailkapen bitar baten aurrean ez gaudenean klase bakoitzak bere doitasun, estaldura eta F1 neurrien balioak izango ditu. Sailkatzaile osoaren ebaluazio bat egiteko horien batezbesteko bat kalkulatzen da. Hiru motatako batezbestekoak dira erabilienak:

- **Makro batezbestekoa:** batezbesteko normala, hau da, klase guztien batura zati klase kopurua:
$$doitasuna_{makro} = \frac{1}{N} \sum_{i=1}^{N} doitasuna_{i\_klasea}$$
- **Mikro batezbestekoa:** kasu honetan metrikaren batezbestekoa egin beharrean, metrika modu global batean aplikatzen da:
$$doitasuna_{mikro} = \frac{\sum_{i=1}^{N} asmatutakoak_{i\_klasea}}{\sum_{i=1}^{N} iragarritakoak_{i\_klasea}}$$
- **Pisatuta:** makro bezalako batezbestekoa baina klase bakoitzari pisu desberdin bat emanez.
$$doitasuna_{pisatuta} = \sum_{i=1}^{N} w_{i\_klasea} \cdot doitasuna_{i\_klasea}$$

2.3.1 Doitasun/estaldura-kurbak

Betiko neurriaz aparte (Doitasuna, Estaldura eta F1) doitasun/estaldura kurbak erabili ditugu. Kurba horiek bi sistema edo gehiagoren arteko portaerak alderatzeko erabiltzen dira. Informazio-erauzketako sistemetan oso erabiliak dira.

Kurba hauek egiteko, beharrezkoa da emaitza bakoitzak konfiantza-pisu bat edukitzea. Horretarako, atalase bat erabiltzen da emaitzen konfiantza-pisuentzat. Atalasea maximotik minimora jaitsiz, sistemak emaitza bakarra ematetik denak ematera pasatzen da, atalasearen balio bakoitzerako doitasuna eta estaldura emanez.



2.3 Irudia: Bi sistemen arteko doitasun/estaldura-kurba baten adibidea.

Hainbat sistema alderatu nahi direnean, bakoitzaren doitasun/estaldura-kurba kalkulatzen da eta denak batera irudikatzen dira. Kurba bat zenbat eta gorago egon planoan, orduan eta eraginkortasun hobea duela esan nahi du. 2.3 irudian adibidez A sistema B baino hobeagoa dela ikus dezakegu. Bi kurbek itxura guztiz desberdina eduki dezaketenez ohikoa da ere kurba hauen azalera kalkulatzea hoberena zein den ikusteko.

3. KAPITULUA

Hizkuntzaren prozesamendurako tresnak eta baliabideak

Atal honetan proiektuan zehar erabilitako baliabideak eta tresnak aipatuko ditugu. Baliabide eta tresna bakoitza zer den deskribatu eta gure proiekturako ze ekarpen/erabilera duen azalduko dugu.

3.1 Baliabideak

Azpiatal honetan erabilitako baliabideak azalduko ditugu. Hauek eskuratzeko helbideak eta proiektuan eduki duten erabilpenaren zehaztasun batzuk aipatuko ditugu.

3.1.1 ConceptNet

ConceptNet [1] hizkuntza anitzeko ezagutza-base bat da. Bertan, eskuz etiketatutako terminoen arteko erlazioak daude adierazita, bai erlazio simetrikoak: *Antonym*, *Synonym*, ... baita asimetrikoak ere: *AtLocation*, *CreatedBy*, *FormOf*, etab.

ConceptNet jasotzen duen hiztegia forma estandarrean dauden terminoek osatzen dute. Termino horiek batzuetan anbiguoak izan arren ez da berezitasunik egiten, horren arrazoi nagusia beste ezagutza edo datu-base batzuekin erlazionatzeko ahalmena izatea da. Hala ere, batzuetan erlazioaren arabera ConceptNetek eskaintzen du desanbiguaziorako baliagarri den informazioa, adieraren kategoria gramatikala, adibidez.

[1] www.conceptnet.io

Ezagutza-basea beste sistema batzuen ezagutzaz hornitzen da ere, WordNet adibidez. Termino bat beste ezagutza-base batetik datorrenean, dakarren erlazioei buruzko informazioaz gain <terminoa, *ExternalUrl*, beste ezagutza-basea> erlazioa erabiltzen da, terminoa eta beste ezagutza-basea lotzeko.

Proiektuaren sarreran azaldu dugun bezala, M. Olney et al. (2012)-en lanean oinarrituta galdera pedagogikoki interesgarriak sortzeko asmoz erlazio gutxi baino interesgarri batzuekin gelditzea erabaki dugu. Aukeratutako erlazioak hurrengoak dira: *AtLocation*, *IsA*, *PartOf* eta *UsedFor*.

3.1.2 WordNet/MCR

WordNet [2] ingelesezko eskuz sorturiko datu-base lexikal handi bat da. Izen, aditz, izenondo eta aditzondoak adiera semantikoen arabera antolatuta daude. Adiera horiei *synset* deritze. Adierak beraien artean konektaturik daude erlazio semantiko eta lexikoen bitartez.

WordNetek duen ezaugarrietako bat fitxategi lexikografoak[3] dira, kategoria sintaktikoan eta taldekatze logikoan daude oinarrituta. Multzo horien adibide batzuk: *act*, *artifact*, *attribute*, *body*, ... dira. Gure proiekturako entitate-izenen kategoriak baino multzo egokiagoak dira.

MCR [4] (Multilingual Central Repository) WordNeten osagarri den datu-base bat da. Bertan ez bakarrik ingelesa, beste bost hizkuntzatara ere hedatuta dago: gaztelera, euskera, katalanera, galiziera eta portugalera. MCRek hainbat proiektu barneratzen ditu, adibidez WordNet Domains [5] (ikusi 4.1.3 azpiatala).

Gure proiektuan batez ere adieren domeinuen informazioa (ikusi 4.1.3. atala) eta fitxategi lexikografoak (ikusi 5.1.2. atala) erabiliko ditugu.

3.1.3 Wikipedia

Wikipedia [6] eduki askeko entziklopedia bat da, lankidetzaz editatua, eleanitza, Interneten argitaratua eta Wikimedia Fundazioak, irabazi asmorik gabeko erakundeak, sustengatua.

[2] www.wordnet.princeton.edu
[3] www.wordnet.princeton.edu/documentation/lexnames5wn
[4] www.adimen.si.ehu.es/web/mcr
[5] www.wndomains.fbk.eu
[6] www.wikipedia.org

Internetera konektatutako edonork parte har dezake Wikipediako artikuluak osatzen.

Hizkuntzaren prozesamenduaren munduan Wikipedia baliabide nagusienetariko bat da. Ez bakarrik doan izateagatik, bere osotasuna eta eguneraketa konstanteak egiten du erakargarri. Hori dela eta, gure corpusa sortzeko erabiliko dugun dokumentu multzoa Wikipediako artikuluak osatuko dute, hain zuzen ere biologia kategoria duten artikuluek. Aipatutako biologia kategoriako artikulu horiek modu automatiko batean erauzi ditugu.

3.2 Software eta liburutegiak

Azpiatal honetan erlazio-erauzlean bai esperimentuetan erabilitako software tresnak eta liburutegiak aipatuko ditugu.

3.2.1 Apache Solr

Apache Solr [7] dokumentuen indexazio eta bilaketarako kode irekiko erreminta bat da. Web bilatzaileetan erabiltzeko pentsatuta dago, hainbat *cluster*-en gainean lan egiteko paraleloan. Zerbitzu hau web API baten bitartez erabiltzen da, html eskaeretan oinarritua.

Solr XML baten bitartez konfiguratzen da, bertan dokumentu bat osatuko duten eremu bakoitza nola indexatuko diren adierazi behar da, hau da, eremuaren mota: zenbakia, karaktere-katea, testua, ... eta mota bakoitzaren konfigurazio bereziak. Testu eremu motaren kasuan adibidez tokenizatzailea, lematizatzailea, eta abar konfiguratu ditzakezu.

```
{
    "indizea" : 7,
    "testua" : "Hau Solr dokumentu baten adibidea bat da.",
    "_version_" : 1574100232473411586
}
```

3.1 Irudia: Solr-eko dokumentu baten formatuaren adibidea, JSON (JavaScript Object Notation) baten parekoa.

Aplikazio honen bitartez gure corpusa indexatu dugu bertan bilaketak azkartzeko.

[7] www.lucene.apache.org/solr

3.2.2 SpaCy

SpaCy [8] hizkuntzaren prozesamendurako Python-eko liburutegi bat da. Beste liburutegi batzuk bezala, tokenizazioa, lematizazioa, erro-erauzketa (stemming), entitate-izenen erazagupena, ... ahalbidetzen du modu erraz batez.

Gure kasuan erraminta hau corpusaren tokenizazio eta lemmatizaziorako erabili da.

3.2.3 Scikit Learn

Scikit-Learn [9] edo SkLearn ikasketa automatikorako Pythoneko liburutegi bat da, NumPy, ScyPy eta matplotlib liburutegietan oinarritua. Ikasketan, ikerketan eta lan munduan oso erabilia da.

Liburutegi honek hainbat gauza eskaintzen ditu: sailkatzaile, erregresio eta clustering algoritmoak, dimentsio murrizketarako algoritmoak, eredu-aukeraketa, aurreprozesaketa, ebaluazio-metrikak, etab.

Proiektu honetan ikasketa automatikoarekin zerikusia duen gehiena liburutegi honen bitartez inplementatua izan da.

3.2.4 Gensim

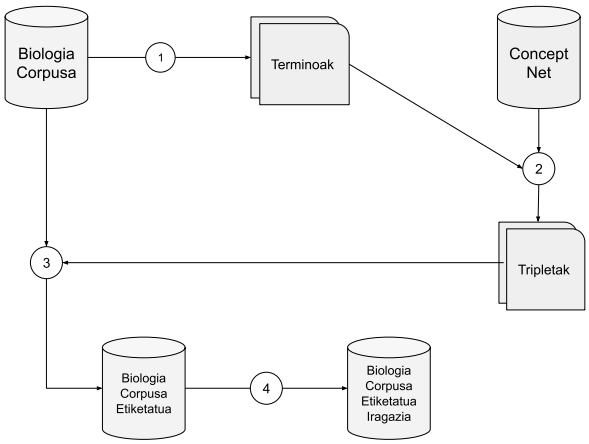
Gensim [10] hizkuntzaren prozesamendurako erremintak eskaintzen dituen Python liburutegi bat da. Zehatzago gai-modelaketa teknikak eskaintzen ditu, besteak beste. Horien artean LDA (Latent Dirichlet Allocation) 4.1.2. atalean aipatuko duguna.

[8] www.spacy.io
[9] www.scikit-learn.org
[10] www.radimrehurek.com/gensim

4. KAPITULUA

Corpusaren etiketatzea urruneko gainbegiraketa erabiliz

Atal honetan azalduko dugu corpusaren etiketaziorako jarraitu dugun prozedura. 4.1. irudian ikus daiteke jarraitu ditugun pausuak. Lehenik eta behin domeinuko terminologia nola erauzi dugun azaltzen dugu. Behin hiztegi bat zehaztuta ConceptNet-eko hirukoteak biologiako domeinura mugatu ditugu. Hurrengo pausuan urruneko gainbegiraketa nola aplikatu dugun azaltzen dugu. Azkenik, teknikak sortutako zarata garbitzeko ideiarekin corpusari aplikatu dizkiogun hainbat iragazketaren aipamena egiten dugu.



4.1 Irudia: Etiketaturiko corpusa lortzeko garapen-lerroaren irudia.

4.1 Termino-erauzketa

Ataza honetan proiektu osoan erabiliko ditugun terminoak definitzen ditugu. Gure proiektua biologiako domeinura mugatu dugunez, azken honi lotuta dauden termino esanguratsuak lortzea da helburua.

Ataza hau betetzeko hainbat metodo probatu ditugu. Alde batetik, maiztasunean oinarrituta TF-IDF neurria, bestetik, gai-modelaketan oinarrituta LDA-Termhood eta, azkenik, WordNet/MCR datu-basean oinarrituta. Ondoren, aipatutako metodoek lortutako terminozerrendak konparatu ditugu eta bat aukeratu dugu.

4.1.1 Termino-maiztasuna - Alderantzizko dokumentu-maiztasuna (TF-IDF)

TF-IDF hitz mailako neurri estatistiko bat da. Hitz bat dokumentu bilduma batean duen garrantzia adierazteko balio du. Neurri honen motibazioa hurrengoa da:

- Hitz bat dokumentu batean askotan agertzen bada, hitzak dokumentu horretan garrantzi handia duela suposa dezakegu. Beraz, hitza duen dokumentuarekiko garrantzia maiztasunarekiko proportzionala dela esan dezakegu.
- Hitz bat dokumentu askotan agertzen bada, adibidez: 'eta', 'edo', ... esanguratsua ez dela pentsa dezakegu, beraz, termino baten dokumentuarekiko garrantzia dokumentu maiztasunarekiko alderantziz proportzionala dela ulertzen dugu.

Aurreko bi suposizioak kontuan hartuta TF-IDF hurrengo moduan definitu dezakegu:

$$TfIdf(t,d,D) = TF(t,d) \cdot IDF(t,D) \tag{4.1}$$

Non t ebaluatu nahi den terminoa adierazten duen, d uneko dokumentua eta $D = (d_1, d_2, \ldots, d_n)$ dokumentu multzoa den. Aipatutako $TF(t,d)$ eta $IDF(t,D)$ funtzioentzako hainbat inplementazio desberdin erabiltzen dira. Orokorrean hurrengoak dira erabilienak:

$$TF(t,d) = \frac{c_{t,d}}{\sum_{t' \in d} c_{t',d}} \tag{4.2}$$

non, $c_{t,d}$ t terminoa d dokumentuan duen agerpen kopuruari egiten dio erreferentzia. Eta

Gaia	Termino esanguratsuenak
0	brain, neuron, neural, cortex, system, nerve, nucleus, motor, sensory, ...
1	system, information, data, database, use, software, computer, program, project, ...
2	gamete, haploid, diploid, allergy, meiosis, produce, zygote, allergic, venom, ...
3	male, female, sperm, mat, sexual, reproductive, offspring, reproduction, mate, ...
4	teeth, dental, tooth, oral, adipose, sport, grade, mouth, massage, ...

4.1 Taula: Egokitutako LDAren gai batzuen termino esanguratsuenak

$$IDF(t,D) = \log \frac{|D|}{|\{d \in D : t \in d\}|} \tag{4.3}$$

TF-IDF neurketa hitz batek dokumentuarekiko duen garrantzia adierazten du. Termino-zerrenda lortzeko dokumentu multzoarekiko duen garrantzia lortu nahi dugunez TF-a kalkulatzeko momentuan kontaketa globala aplikatu dugu.

4.1.2 LDA eta Termhood

Termino-erauzketan emaitzak hobetzearren, Li et al. (2013)-ek proposatutako *Termhood* neurria erabili dugu. Horrek aldi berean Blei et al. (2001) artikuluan proposatutako LDAn (Latent Dirichlet Allocation) oinarritzen da.

LDA gai-modelaketa atazan kokatzen da. Ataza horren helburua testu batean ager daitezken gaiak erauztea da. Metodo honek eredu probabilistikoetan oinarritzen da testuko gaien distribuzioa eta gai hauek errepresentatzen duten hitzen distribuzioa estimatzeko (ikusi 4.1. taulako adibideak).

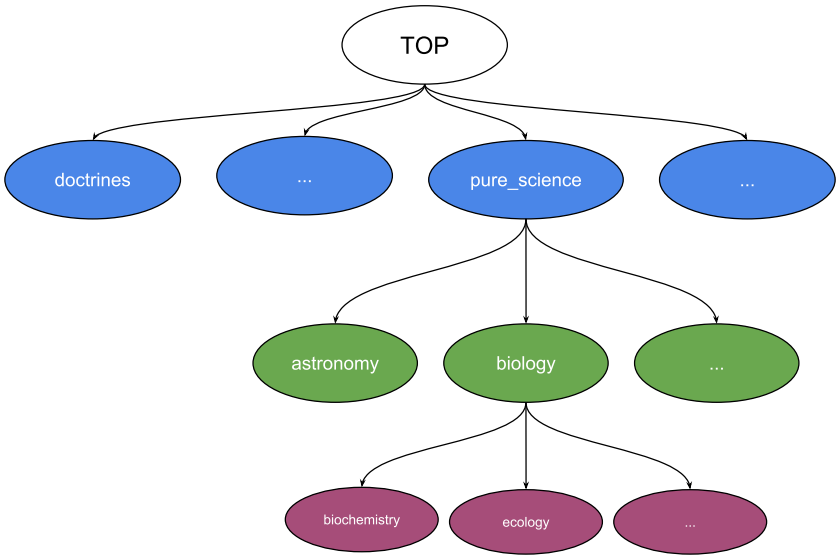
Behin LDA modeloa corpusaren gainean egokitu dugula, *Termhood* neurria honela definitzen dugu:

$$Termhood(t) = \log(TF(t)) \cdot \phi_t^{mt_t}$$

$$mt_t = \arg\max(\phi_t^{T_i})$$

non $\phi_t^{T_i}$ T_i gaia definitzen duen hitz distribuzioan t terminoa duen probabilitatea den. t terminoa mt_t gaiarekiko duen probabilitatea beste gai guztiena baino handiagoa den gaia da mt_t. Eta $\phi_t^{mt_t}$ t terminoa mt_t gaiarekiko duen probabilitatea da.

Domeinu bat gai orokor bat bezala ulertuta, eta gaia azpigaiez osatuta dagoela jakinda,



4.2 Irudia: WordNet Domains-en hierarkia erakusten duen irudia.

azpigai horientzako esanguratsuak diren hitzak domeinurako ere esanguratsuak izatea espero dugu.

Esperimentaziorako hurrengo azpigai kopuruekin egin dugu proba: 25, 50 eta 100. Ondorioztatu dugu daukagun dokumentu multzoarekin LDA eredua 100 azpigai bereizteko gaitasuna duela (ikusi 4.1. taulan azpigaien adibide batzuk.).

4.1.3 WordNet/MCR

WordNeten adiera bakoitza (ikusi 3.1.2. atala) WordNet Domainseko domeinu bati lotuta dago. WordNet Domains hierarkikoki antolatuta dauden 200 domeinu baino gehiagoz dago osatuta (ikusi 4.2. irudia).

Terminoen zerrenda osatzeko, bai biologia domeinuko baita biologia domeinuaren azpitik dauden beste domeinuetako adierak hartu ditugu, guztira **41034** terminoz osaturiko zerrenda osatuz.

4.1.4 Emaitzak

Atal honetan termino-erauzketan lortutako emaitzak aipatuko ditugu. Aurreko ataletan erakutsitako planteamenduen alde onak eta txarrak aurkeztuko ditugu, eta hartutako erabakia zein den eta zergatik adieraziko dugu.

TF-IDF	LDA/Termhood	WordNet/MCR
Plant	Virus	Living thing
Cell	Bird	Animate thing
Protein	Language	Organism
Use	Medical	Being
Specie	Blood	Benthos
Also	Male	Dwarf
Gene	Female	Heterotroph
Pathogen	Research	Parent
Disease	Health	Life
Virus	Bacteria	Biont
...	...	...

4.2 Taula: Metodo desberdinen bidez lortutako hitz-zerrenden hasierako hitzen konparaketa

4.2. taulan ikus ditzakegu hiru metodoak erabiliz lortu ditugun hasierako hamar terminoak. TF-IDF eta Termhood-aren kasuan balio handieneko terminoak erakusten ditugu eta WordNet/MCRren kasuan, berriz, ausaz aukeratutakoak.

TF-IDF zutabera begiratzen badiogu **Use** edo **Also** motako hitzak aurkitzen ditugu. Hitz horiek ez dira biologiako domeinukoak, beraz, hitz zaratatsuak bezala kontsideratzen ditugu. Ikusten dugun bezala, maiztasunean soilik oinarritutako metodoak ez dira gai oso ohikoak diren hitz hauek ekiditen.

LDA/Termhood neurria aipatutako hitz zaratatsuen arazoa konpontzen saiatzen da maiztasunari garrantzia txikituz. Ikus daiteke nola TF-IDF metodoak aurkezten duen zarata desagertu egiten dela. LDAren informazioaz baliatuta emaitzak nahiko hobetzen direla argi gelditzen da; hala ere, metodoa corpusaren guztiz menpekoa denez, zuzenean biologiarekin erlazionatuta ez dauden beste termino batzuk agertzen dira, **Language** edo **Medical** adibidez.

Azkenik, WordNet/MCR-k, zehatzago WordNet Domains, eskuz etiketaturiko zerrenda bat denez, bertako elementu guztiak biologiako terminoak direla ziurtatzen digu.

Hiru metodoak konparatu ondoren WordNet/MCRrekin lortutako termino-zerrenda erabiltzea erabaki dugu. Arrazoi nagusia beste metodoek erakusten dituzten arazoak aurkez-

```
/a/[/r/Antonym/,/c/en/ophiophobia/n/,/c/en/ophiomania/]
/r/Antonym
/c/en/ophiophobia/n
/c/en/ophiomania
{
    "dataset": "/d/wiktionary/en",
    "license": "cc:by-sa/4.0",
    "sources": [
        {
            "contributor": "/s/resource/wiktionary/en",
            "process": "/s/process/wikiparsec/1"
        }
    ],
    "weight": 1.0
}
```

4.3 Irudia: ConceptNeteko sarrera baten adibidea.

ten ez dituelako da; hau da, TF-IDF metodoak dituen hitz zaratasuak eta LDA/Termhood dituen domeinutik kanpoko hitzak. WordNet/MCR metodoarekin lortutako zerrenda erabiltzearen beste abantaila bat da jadanik termino bakoitzaren adiera identifikatuta edukitzea, gero, terminoen gaineko informazio gehigarria eskuratzeko prozesua erraztuz.

4.2 ConceptNet iragazketa

Atal honetan erabiliko dugun ezagutza-basea, ConceptNet (ikusi 3.1.1. atala), gure domeinura murrizteko aplikatutako prozesua azalduko dugu. Prozesua osatzen duten pausoekin batera, pauso bakoitza deskribatzen duen taula bat aurkeztuko dugu. Taula horretan hiru neurri agertuko dira, hain zuzen ere, erlazio bakoitzaren hirukote kopurua, erlazio bakoitzaren proportzioa ezagutza-basean eta erlazio bakoitza duen termino estaldura, hau da, ezagutza-basean agertzen diren termino guztietatik zenbatek hartzen dute erlazio horretan parte. ConcetpNet iragazteko prozesuan hiru pauso bereizten dira:

1. **Ingelesezko ConceptNeta lortzea:** ConceptNet hizkuntza anitzeko ezagutza-base bat denez lehendabiziko pausoa guk erabiliko dugun hizkuntzara murriztea da. Lehenengo iragazketa hau aplikatzeko ConceptNetek eskaintzen duen informazioaz baliatuko gara, informazio horren artean hirukote bakoitza osatzen duten entitate

Erlazioa	Kopurua	Proportzioa	Termino estaldura
RelatedTo	1586448	0.5025	0.5067
FormOf	353457	0.1119	0.5024
IsA	229071	0.0725	0.1380
Synonym	218995	0.0693	0.1682
HasContext	207387	0.0656	0.1376
DerivedFrom	177275	0.0561	0.1690
ExternalURL	57868	0.0183	0.0504
UsedFor	39790	0.0126	0.0286
...	...	...	...
AtLocation	27746	0.0087	0.0148
...	...	...	...
PartOf	12988	0.0041	0.0109
...	...	...	...

4.3 Taula: Hasierako ConceptNeteko azterketa: beltzez hautatutako erlazioak.

bikoteen hizkuntzari buruzko informazioa hain zuzen ere (ikusi 4.3. irudia). Behin ingelesezko hirukoteak identifikatuta beste guztiak ezabatu ditugu.

Lortutako ConceptNeteko azpimultzoko neurketa batzuk erakusten dira 4.3. taulan. Taulari erreparatuz *RelatedTo* da erlazio nagusia ezagutza-basearen **%50** izanda. Ikusten dugu ezagutza-basean orokorrean desoreka oso handia dagoela erlazioen artean.

Pauso honetan ere formatu trinkoago bat eman diogu ezagutza-baseari. Formatu berri horretan beharrezkoa den informazioa bakarrik utzi dugu, hau da, 4.3. irudiko adibidea <*Ophiophobia*, *Ophiomania*, *Antonym*> bezala utzi dugu.

2. **ConceptNet domeinura mugatzea:** ConceptNet domeinu irekiko ezagutza-base bat denez biologiako domeinutik kanpo dauden hirukoteak ezabatu behar ditugu. Ezagutza-baseko hirukote bat domeinukoa izango da baldin eta hirukotea osatzen duten entitate bikotea domeinuko termino zerrendan (4.1. atalean azaldutakoa) agertzen badira. Iragazketa honetan bi aukera aztertu genituen: alde batetik entitate bikotearen bi entitateetako batek domeinuko termino zerrendan agertzea, eta bestetik, entitate bikotearen bi entitateek termino zerrendan agertzea. Azkenik bigarren aukerarekin gelditu ginen, hau da, entitate bikotea termino zerrendan agertzea. Aukeraketa honen arrazoia hirukote zaratasuak ekiditea da.

   4.4. taulari erreparatzen badiogu ikusten dugu hirukote kopuru handiko erlazio batzuk, *FormOf* adibidez, hirukote kopuruan nahiko jaitsi direla. Orokorrean esan

Erlazioa	Kopurua	Proportzioa	Termino estaldura
RelatedTo	33927	0.4368	0.5463
IsA	17105	0.2202	0.6094
Synonym	14395	0.1853	0.7397
HasContext	3213	0.0413	0.1386
DerivedFrom	2955	0.0380	0.1943
PartOf	1881	0.0242	0.0931
FormOf	647	0.0083	0.0543
AtLocation	543	0.0069	0.0191
...	...	...	...
UsedFor	159	0.0020	0.01
...	...	...	...

4.4 Taula: Domeinuko ingelesezko ConceptNeteko azterketa: beltzez hautatutako erlazioak.

Erlazioa	Kopurua	Proportzioa	Termino estaldura
IsA	17105	0.8688	0.9895
PartOf	1881	0.0955	0.1512
AtLocation	543	0.0275	0.0311
UsedFor	159	0.0080	0.0163

4.5 Taula: Domeinuko ingelesezko ConceptNeteko azterketa: erabiliko ditugun erlazioak bakarrik kontuan hartuta.

dezakegu aurretik geneukan desoreka pixka bat leundu dela.

3. **Erlazioak aukeratzea:** Pauso honetan aldez-aurretik erabakitako erlazioak besterik ez ditugu utziko ezagutza-basean. Gure problemarako erakargarriak izan daitezkeen erlazioetara mugatu dugu ezagutza-basea, erlazio horiek: *AtLocation*, *IsA*, *PartOf* eta *UsedFor* dira.

   4.5. taulan ikus ditzakegu ezagutza-basearen azkeneko emaitzak. Lortu dugun ConceptNeta oso desorekatua dago, adibidez, *IsA* erlazioak *UsedFor* baino ehun aldiz adibide gehiago ditu. Desoreka honek hurrengo pausoetan lortuko ditugun emaitzetan eragina izango du.

4.3 Urruneko gainbegiraketaren aplikazioa

Behin gure ezagutza-basea murriztuta dugula corpusa etiketatzeko momentua da. Hori egiteko 2.1 atalean ikusi dugun teknika, hau da, urruneko gainbegiraketa erabiliko dugu.

Teknika hau aplikatzeko ezagutza-baseaz gain dokumentu multzo bat ere behar dugu. Dokumentu multzoa lortzeko ingelesezko Wikipediatik **biologia** kategoria duten artikuluak erauzi ditugu.

Atal honetan azalduko dugu corpusaren etiketazioa urruneko gainbegiraketa erabiliz aurrera eramateko jarraitu ditugun pauso guztiak.

4.3.1 Corpusaren prozesaketa eta indexazioa

Azpiatal honetan azalduko dugu Wikipediako artikulu multzotik corpusa osatuko duten dokumentuetara iristeko eman ditugun pausoak. Gure problema esaldi mailan gertatzen da, beraz, lehendabiziko pausoa artikulu multzoa esaldi multzora zatitu dugu. Esaldi bakoitzari dokumentua deituko diogu eta dokumentu bakoitzari **docid** identifikadore bat esleitu diogu.

Aurreprozesaketa
Lan egin ahal izateko hainbat prozesaketa aplikatu dizkiegu dokumentuei. Lehendabiziko eta oinarrizkoena **tokenizazioa** izan da. Tokenizazioa dokumentua token deituriko elementuetan zatitzean datza; tokenak hitzak, puntuazio markak, ... izan daitezke.

Tokenizazioaz aparte **lematizazioa** ere aplikatu dugu. Lematizazioa token bakoitzeko oinarrizko forma lortzean datza, lema izenekoa. Adibidez, 4.6 taulan *is* tokena *be* lema bihurtzen da. Lematizazioa oso interesgarria da, batez ere, termino baten agerpena bilatu behar dugunean.

Azkenik token bakoitzeko **kategoria gramatikala** erauzi dugu. Kategoria gramatikalak token bakoitza izen, aditz, adjetibo, aditzondo, ... den adierazten du. Prozesuan geroago erabiliko dugun informazioa eskaintzen digu, hain zuzen ere, interesgarri izan daitezkeen ezaugarri sintaktikoak erauzi ahal izateko informazioa.

Prozesaketa horiek guztiak SpaCy liburutegiaren bitartez egin ditugu (ikusi 3.2.2. atala).

Indexazioa

Urruneko gainbegiraketa aplikatzeko gure dokumentu multzoan terminoen agerpenak bilatu behar ditugu. Konputazionalki oso garestia denez, bilaketa-motore bat erabiltzea erabaki dugu. Proiektu honetarako **Apache Solr** aukeratu dugu (ikusi 3.2.1. atala).

Indexatzeko momentuan bi osagaiz osaturiko erregistroak definitu ditugu, alde batetik, dokumentua bera **text** izenarekin, eta, bestetik, dokumentua identifikatuko duen **docid** iden-

Esaldia	Agricultural science is a broad multidisciplinary field that encompasses the parts of exact, natural, economic and social sciences that are used in the practice and understanding of agriculture.
Tokenak	agricultural science is a broad multidisciplinary field that encompasses the parts of exact , natural , economic and social sciences that are used in the practice and understanding of agriculture .
Lemak	agricultural science be a broad multidisciplinary field that encompass the part of exact , natural , economic and social science that be use in the practice and understanding of agriculture .
Kategoria gramatikalak	JJ NN VBZ DT JJ JJ NN WDT VBZ DT NNS IN JJ , JJ , JJ CC JJ NNS WDT VBP VBN IN DT NN CC NN IN NN .

4.6 Taula: Dokumentu baten prozesaketaren ondorioz lorturiko emaitzak.

tifikadorea. Testuaren indexazio-konfiguraziorako Solr-ek dakarren **text_general** mota erabiltzea erabaki dugu. Indexatu dugun testua lematizatutako testua izan da.

4.3.2 Ikasketarako adibideen erauzketa

Azpiatal honetan corpusa osatuko duten adibideen erauzketa-prozesua azalduko dugu. Alde batetik, 4.2 atalean lortutako hirukoteen adibideak, hau da, adibide positiboak. Bestetik, adibide negatiboak edo erlazio gabeko adibideak (aurrerago NIL deiturikoak).

Adibide positiboen erauzketa

Adibide positiboak urruneko gainbegiraketaren hipotesia betetzen duten esaldiak dira, hau da, ezagutza-baseko hirukote baten bi argumentuak jasotzen dituzten esaldiak. Esaldi horiek aurkitzeko gure dokumentu multzoan argumentuen agerpenak begiratu behar ditugu. Lortuko dugun erlazioekiko adibide kopurua erlazioekiko hirukote kopuruaren proportzionala izatea espero dugu.

Adibide negatiboen (NIL) erauzketa

Adibide negatiboak ezagutza-basean agertzen ez diren hirukoteen dokumentu multzoko agerpenak osatzen dituzte. 2.1.1 atalean azaltzen dugun bezala, Mintz et al. (2009)-ek erabilitako estrategia jarraitu dugu, hau da, aukeratu ezagutza-baseko hirukoterik osatzen ez dituzten argumentu pare bat ausaz eta horien agerpenak testuan bilatu. Corpus oso zaratatsua ez lortzearren NIL adibide kopurua beste klaseen antzeko adibide kopuru bat erabiltzea erabaki dugu.

Erlazioa	Adibide kopurua	Proportzioa
AtLocation	13988	0.05
IsA	172125	0.66
PartOf	51821	0.19
UsedFor	12558	0.04
NIL	10000	0.03

4.7 Taula: Corpusa osatzen duten adibideen kopurua erlazioen arabera.

1. entitatea	2. entitatea	Erlazioa	Esaldia
mosses	bryophytes	IsA	The **bryophytes**, which include liverworts, hornworts and **mosses**, reproduce both sexually and vegetatively.
hair	head	AtLocation	When the **head** is in a normal upright position, the otolith presses on the sensory **hair** cell receptors.

4.8 Taula: Corpusetik ateratako adibide batzuk.

4.3.3 Emaitzak

4.7 taulan ikus dezakegu erlazio bakoitzarentzat zenbat adibide lortu ditugun. Adibide kopurua 4.5 taularekin alderatzen badugu ikus dezakegu hirukote bakoitzeko **21.9** adibide lortu ditugula batezbesteko **311.3** adibideko desbiderapenarekin. Azkenik aipatzeko datumultzoan gehien agertzen den hirukote bakoitzeko adibide kopurua **1** dela esan beharra dago. Emaitza hauek NIL klaseko adibideak kontuan hartu gabekoak dira. Hala ere, antzeman dezakegu nola hasieran genuen desoreka arazoa pixka bat orekatu den, adibidez, *IsA* klasea **%82**tik **%66**ra jaitsi da.

Erauzitako adibideei erreparatzen badiegu bai egokiak baita zaratatsuak ere aurkitzen ditugu, 4.8. taulan ikus dezakegun bezala. Lehenengo adibidean ikus dezakegu lortutako esaldia benetan erlazioa erakusten duela, kontrajarrian, bigarrenak ez du erlaziorik erakusten. Zaratatsuak diren adibide guztiak identifikatzeko eskuzko lan baten beharra legoke, lan hori oso garestia denez, hurrengo atalean azaltzen dugu modu heuristiko batean arazo horri aurre egiteko aplikatutako metodo batzuk.

4.4 Corpusaren iragazketa

Atal honetan azalduko dugu gure corpusean agertzen diren adibide zaratatsuak garbitzen saiatzeko egindako prozesua. Intxaurrondo et al. (2013)-ek erakusten dute nola heuristiko batzuen bitartez corpusean agertzen diren adibide zaratatsuak nola garbi daitezkeen. Guk jarraitutako prozesua beraiena bezalakoa zeharo ez izan arren antzeko heuristikoak erabili ditugu. Aipatu beharra dago egin ditugun iragazketa guztiak hirukote mailan izan direla.

4.4.1 Argumentuen antzekotasunean oinarritutako iragazketak

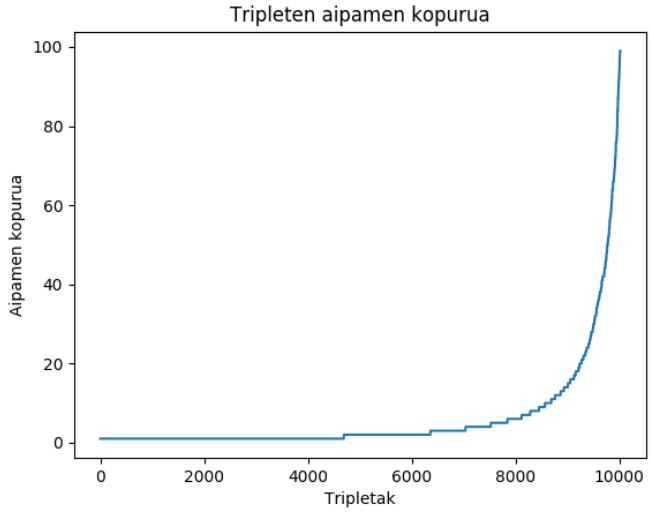
Hurrengo iragazketak hirukoteen argumentuetan oinarritzen dira, bi motakoak bereizi ditugu. Alde batetik, hirukote baten bi argumentuak berdinak direnean ezabatzea, eta, bestetik, hirukote bateko argumentu bat beste argumentuan agertzea azken honek hitz anitzeko unitate bat izanda. Iragazketa hauek <Bird, Bird, PartOf> edo <Cell membrane, Membrane, IsA> moduko hirukoteak ezabatzeko balio digu, lehenengoaren kasuan hitz baten bi adiera osatzen dutelako eta horren adibide onak lortzea oso zaila delako, eta, bigarrenaren kasuan, hirukote erredundanteak direlako.

Mota horietako hirukoteak orokorrean beste arazo bat sortzen dute adibideak erauzteko momentuan, adibidez, gerta daiteke esaldi batean "Cell membrane"argumentua aurkitzea, eta ondorioz, "membrane"ere beti aurkituko da. Kasu horietan adibideek ez dute "IsA"erlazioa erakutsiko eta, beraz, zaratatsuak izango dira.

4.4.2 Adibide kopuruan oinarritutako iragazketak

Iragazketa honek hirukote batek corpusean duen agerpen kopurua du kontuan. Hirukote batek geroz eta aipamen kopuru gehiago izan orduan eta zaratatsuagoa izateko probabilitate handiagoa dauka. Ikusi dugu 4.3.3 ataleko emaitzetan desbiderapen handi bat dagoela aipamen kopuruen artean, hori dela eta, hirukoteen aipamen kopurua irudikatu dugu 4.4 irudian. Irudian ikus daiteke hirukote bakoitzaren aipamen kopurua (gehienez 100 aipamen dituztenak bakarrik), aipamen kopuruaren arabera.

Eskuzko azterketa baten ondoren adibide bakar bat duten hirukoteek orokorrean adibide onak dituztela ikusi dugu, aldiz, ehun baino adibide gehiago dituzten hirukoteak, ordea, jasotzen dituzte adibide zaratatsu gehienak. Beraz, 100 aipamen baino gutxiago dituzten hirukoteekin gelditu gara.



4.4 Irudia: Hirukoteen agerpen kopurua erakusten duen grafikoa, agerpen kopuruen arabera ordenatuta.

Iragazketa hau hirukoteen %1 bakarrik ezabatu arren adibideen %46.79 ezabatzen ditu. Hirukote horien artean badaude hirukoteak 1800, 1400, 1000, ... aipamenekin. Enpirikoki ikusi dugu hirukote hauek iragazita emaitzak hobetu egiten direla.

4.4.3 Elkarrekiko informazio puntuala

Elkarrekiko informazio puntuala (Pointwise Mutual Information ingelesez) bi argumentu independenteren artean, argumentuek batera agertzea datu-multzoan informatiboa den edo ez adierazten digu. Heuristiko honek testuinguru okerra izan dezaketen aipamen zaratatsuetan jartzen du arreta.

Hipotesiaren arabera, argumentuak korrelatuta egongo dira baldin eta argumentu bikote bat askotan azaltzen bada aipamen desberdinetan, eta gutxiagotan independenteki. Bestalde, argumentu pare baten agerpen maiztasuna, bi argumentuak indibidualki agertzeko duten maiztasuna baino askoz baxuagoa bada, orduan aipamen horiek zaratatsuak direla pentsa dezakegu.

Neurketa hau Min et al. (2011)-en lanean dago oinarrituta, hurrengo formularekin definituta:

$$pmi(arg_1, arg_2) = \log \frac{p(arg_1, arg_2)}{p(arg_1)p(arg_2)} \tag{4.4}$$

$p(arg_1, arg_2)$ bi argumentuak adibide berean agertzeko duten probabilitatea da, eta $p(arg_i)$ i. argumentua adibide batean agertzeko duen probabilitatea.

Behin argumentu pare bakoitzaren PMI balioa lortuta, atalase baten azpitik dauden hirukoteak eta beraien adibideak ezabatzen ditugu. Iragazketa bakarrik aipamen positiboen gainean aplikatu dugu.

Gure esperimentazioaren arabera **1.5** baliotik behera dauden adibideak kenduta lortu ditugu emaitzarik onenak.

4.4.4 Emaitzak

Iragazketen ondoren gelditzen zaigun corpusak dituen proportzioak 4.9 taulan erakusten ditugu.

Erlazioa	Adibide kopurua	Proportzioa
AtLocation	4655	0.08
IsA	34959	0.59
PartOf	8044	0.13
UsedFor	858	0.01
NIL	10000	0.17

4.9 Taula: Iragazitako corpusa osatzen duten adibideen kopurua erlazioen arabera.

Prozesu honetan bi helburu lortu ditugu nagusiki, alde batetik, erlazioen arteko adibide distribuzioa hain alboratsua ez izatea, eta, bestetik, zaratatsu izan daitezken hainbat adibide iragaztea.

Distribuzio desorekaren kasuan ikus dezakegu 4.7 eta 4.9 taulak alderatuta, *IsA* beste klaseetara gerturatzea lortu dugula, *UsedFor* adibide asko galdu arren. Orokorrean erlazio guztien adibide kopurua nahiko jaitsi da, corpusaren tamaina nabarmenki jaitsiz.

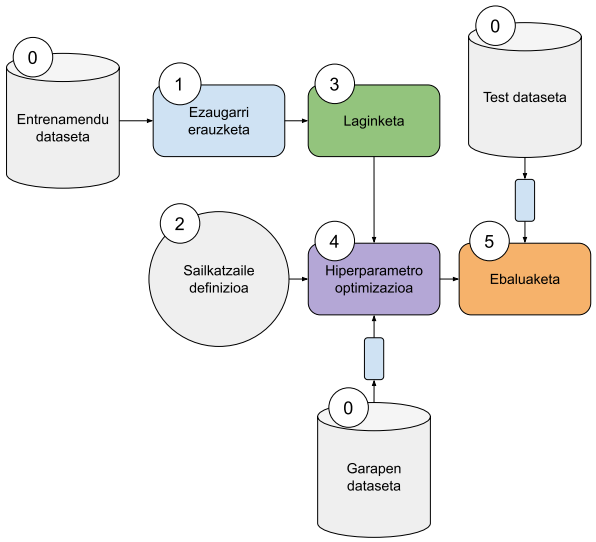
Iragazketen ebaluaketa egiteko, lehenik eta behin 5.1.1. atalean azaldutako entrenamendu, garapen eta test datu-multzo zatiketa egin dugu; sortutako datu-multzo berriek hasierako corpusaren %70, %20 eta %10 proportzioak jarraitzen dituzte urrenez-urren. Behin zatiketa aplikatuta 5. atalean azalduko dugun sistema erabili dugu, modu iteratibo batez bi aldeetan, bai corpusean baita sailkatzailean ere, hobekuntzak eginez. Ikusi dugu nola batez ere, *IsA* erlazioak sortzen dituen faltsu-positiboak iragazketen ondoren nahiko jaitsi direla.

Eskuz etiketaturiko test datu-multzorik ez genuenez, hau da, automatikoki sortzen ditugun test datu-multzoak zaratatsuak direnez, ezin izan dugu iragazketek lortzen dituzten hobekuntza zuzenean neurtu. Hala ere, aurreko lanetan (Intxaurrondo et al. (2013)) eta gure sistemak lortutako emaitzen hobekuntzetan oinarrituta, aipatutako iragazketak justifikatuta daudela ondorioztatzen dugu.

5. KAPITULUA

Erlazio-erauzlearen sorkuntza eta ebaluazioa

Hurrengo azpiataletan erlazio-erauzle sistemaren sorkuntza eta ebaluazioari buruz jarraitu ditugun pausoak aipatuko ditugu. Alde batetik, diseinuan eta inplementazioan hartutako erabakiak eta erabilitako algoritmoak azalduko ditugu eta, bestetik, sistemaren ebaluazioan lortutako emaitzak komentatuko ditugu.



5.1 Irudia: Erlazio-erauzlearen inplementazioan jarraitutako prozesua.

Datu-multzoa	Adibide kopurua	Proportzioa
Entrenamendukoa	40966	0.7
Garapenekoa	11701	0.2
Testekoa	5849	0.1

5.1 Taula: Datu-multzo berrien adibide kopurua eta proportzioak

5.1 Erlazio-erauzlearen inplementazioa

Azpiatal honetan erlazio-erauzlearen inplementazioan jarraitu ditugun pausoak azalduko ditugu (ikusi 5.1 irudia). 0. pauso bezala entrenamendu, garapen eta testerako datu-multzoko zatiketa aplikatu dugu (ikusi 5.1.1. atala). Lehenengo pauso bezala ezaugarri batzuk definitu ditugu (ikusi 5.1.2. atala), gero sailkatzaile aukeraketa prozesu bat jarraitu dugu (ikusi 5.1.3. atala). Hirugarren pausotzat laginketa metodoak aplikatu dizkiogu entrenamendu datu-multzoari erlazioen arteko distribuzioan oreka bat lortzeko (ikusi 5.1.4). Ondoren, aukeratutako sailkatzailearen hiperparametroak optimizatu ditugu garapen datu-multzoa erabiliz (ikusi 5.1.5. atala). Azkenik, sistema ebaluatu dugu testerako datu-multzoa erabiliz (ikusi 5.2)

5.1.1 Entrenamendu, garapen eta test datu-multzoak

Esperimentazioa zentzuzkoa izan dadin gure corpusa hiru azpimultzotan banatu dugu; entrenamendu, garapen eta test datu-multzoak. Entrenamenduko datu-multzoa izango da erabiliko duguna gure sailkatzailea entrenatzeko; egingo ditugun moldaketa guztiak honen gainean egin ditugu. Garapen datu-multzoa erabiliko dugu bai hiperparametroak aukeratzeko baita bestelako erabakiak hartzeko ere. Azkenik, guztiz objektiboak izateko, orain arte ikusi ez dugun test datu-multzoa erabiliko dugu erlazio-erauzlea ebaluatzeko.

Datu-multzo hauek 4. kapituluan aipatutako prozesuan lortutako azken datu-multzoaren azpimultzoak dira. Azpimultzoak sortzeko momentuan hirukote baten adibide guztiak azpimultzo berean egotea bermatu dugu. Entrenamendu, garapen eta test azpimultzoen tamaina originalarekiko %70, %20 eta %10 dira hurrenez hurren (ikusi 5.1. taula).

5.1.2 Ezaugarrien erauzketa

Ikasketa-automatikoko algoritmoak ez dira gai testuaren gainean zuzenean lan egiteko. Hori dela eta, testutik ezaugarriak erauzi eta algoritmoari pasako dizkiogu bere sailkape-

Esaldia
Some organisms are made up of only one cell and are known as unicellular organisms.

5.2 Irudia: <*Cell*, *Organism*, *PartOf*> hirukotearen agerpena.

Ezk. leihoa	FLI1	Tarteko hitzak	FLI2	Esk. leihoa
	Tops	are/VBP made/VBN up/RP of/IN only/RB one/CN	Tops	
some/DT	Tops	are/VBP made/VBN up/RP of/IN only/RB one/CN	Tops	and/CC
some/DT	Tops	are/VBP made/VBN up/RP of/IN only/RB one/CN	Tops	and/CC are/VBP

5.2 Taula: Mintz et al. (2009)-ek proposaturiko ezaugarri lexikalen adibide bat.

nak egiteko. Proiektu honetan Mintz et al. (2009)-ek eta Zhou et al. (2005)-ek proposatutako ezaugarria erabili ditugu. Ezaugarrien adibideak erakusteko 5.2. irudiko esaldian oinarrituko gara.

Hasiera bateko ezaugarriak

Mintz et al. (2009)-ek proposaturiko erabili ditugu oinarri-lerrotzat. Artikuluan aipatutako sistema adibideen testuinguruaren ezaugarri lexikal eta sintaktikoetan daude oinarrituta. Gure kasurako bakarrik lexikalak hartu ditugu kontuan gauzak sinplifikatzearren.

Ezaugarri bakoitzak hurrengo informazioa erakusten du:

- Zein entitate datorren lehendabizi erakusten duen etiketa.
- Bi entitateren arteko hitzak eta hitz hauen kategoria gramatikala.
- Lehendabizi datorren entitatearen aurretik dauden k hitzak eta hitz hauen kategoria gramatikala.
- Bigarren datorren entitatearen ondoren dauden k hitzak eta hitz hauen kategoria gramatikala.

Ezaugarri bakoitza aurreko osagai guztien bateraketa bat da $k \in 0,1,2$ balio desberdinetarako (ikusi 5.2. taula). Mintz et al. (2009)-ren lanean argumentu bakoitza beraien entitate-izenen kategoriaren (named entity category) ordez aldatzen dira, gure kasuan, ordea, domeinurako egokiagoak diren fitxategi lexikografoen izenak erabili ditugu.

Hasiera bateko ezaugarri hauek erabilita lortu ditugun emaitzak (5.2.1. atalean azalduta) ez dira oso onak izan. Arrazoi nagusia, seguruenik, ezaugarri oso luzeak direlako da. Horregatik, ezaugarri hauek duten estaldura oso txikia da, eta, beraz, ez dira oso erabilgarriak. Hori dela eta, beste ezaugarri batzuk erabiltzea erabaki dugu.

Sistemaren ezaugarriak

Hemen proposaturiko ezaugarriak Zhou et al. (2005)-en lanean aipaturiko ezaugarrietan oinarrituta daude. Aurreko ezaugarriekin konparatuz, informazio berdina jasotzen dute baina modu zatikatuago batean, ezaugarri zehatzagoak erabiliz. Zati bakoitza zer informazio adierazten duen erakusteko hurrengo etiketak erabiltzen dira:

- *PF*: Zein entitate datorren lehenengo adierazten duen etiketa.
- *WM1*: lehenengo entitatea osatzen duten hitzak.
- *HM1*: lehenengo entitatearen burua.
- *WM2*: bigarren entitatea osatzen duten hitzak.
- *HM2*: bigarren entitatearen burua.
- *HM12*: *HM1* eta *HM2*ren konbinaketa.
- *WBNULL*: bi entitateen artean hitzik ez dagoenean.
- *WBFL*: bi entitateen artean dagoen hitza bi entitateen artean hitz bakarra dagoenean.
- *WBF*: bi entitateen arteko lehendabiziko hitza bi entitateen artean bi hitz edo gehiago daudenean.
- *WBL*: bi entitateen arteko azkeneko hitza bi entitateen artean bi hitz edo gehiago daudenean.
- *WBO*: bi entitateen arteko hitzak, lehendabizikoa eta azkenekoa ezik, bi entitateen artean hiru hitz edo gehiago daudenean.
- *BM1F*: lehenengo entitatearen aurretik dagoen lehenengo hitza.
- *BM1L*: lehenengo entitatearen aurretik dagoen bigarren hitza.
- *AM2F*: bigarren entitatearen ondoren dagoen lehenengo hitza.
- *AM2L*: bigarren entitatearen ondoren dagoen bigarren hitza.
- *ET12*: bi entitateen fitxategi lexikografoen izenen batura.

Etiketa	Balioa
PF:	1
WM1:	organisms
HM1:	organisms
WM2:	cell
HM2:	cell
HM12:	organisms cell
WBF:	are
WBL:	one
WBO:	made up of only
BM1F:	some
AM2F:	and
AM2L:	are
ET12:	Tops Tops

5.3 Taula: Zhouk proposaturiko ezaugarrien adibide bat.

5.1.3 Sailkatzailea

Gure problema ebazteko erabiliko dugun algoritmoa aukeratzeko, hainbat hautagairen arteako konparaketa egin dugu: *Naive Bayes* ereduak, erabaki zuhaitza motakoak, eredu linearrak eta Sostengu Bektore Makina motakoak. Azkenean probak egin ondoren Sostengu Bektore Makinak (ikusi 2.2. atala) erabiltzea erabaki dugu.

Sailkatzailearen inplementaziorako Scikit-Learn liburutegia erabili dugu. Pythoneko liburutegi honek atzetik duen inplementazioa *C* lengoaian idatzitako LIBSVM liburutegiaz baliatzen da eraginkortasua bermatzeko. Proiektu osoan erabili den kodea Pythonez idatzi dugunez egokia eta erosoa ikusi dugu liburutegi hau erabiltzea.

5.1.4 Laginketa

Aurretik aipatu dugun bezala (ikusi 4. kapituluan) gure sailkatzailea entrenatzeko erabiliko dugun datu-multzoa desorekatuta dago. Desorekak ekar dezakeen arazo nagusia sailkatzailea adibide gehien dituen klasea bakarrik sailkatzea edo, beste era batean esanda, adibide gutxi dauzkaten klaseko adibideak ondo ez sailkatzea da.

Hori dela eta, aipatutako arazoa saihesteko hainbat proposamen kontsultatu ditugu. Proposamen hauek adibide sintetikoen sorkuntzan oinarritzen diren teknikak dira. Lan honetan **SMOTE** eta **Tomek Links** algoritmoak erabili ditugu batera. Lehenengoa, adibide sinte-

tiko berriak sortzeko erabiliko dugu, eta, bigarrena, nahasturik egon daitezkeen adibideak ezabatzeko.

SMOTE algoritmoa

SMOTE (Synthetic Minority Over-sampling Technique) (ikusi 1. algoritmoa)Bowyer et al. (2011)-ek proposatutako adbide sintetikoak sortzeko algoritmoa da. Normalean, laginketa bat aplikatu behar denean datu gordinen gainean lan egiten da; adibidez, irudien kasuan irudi baten antzekoa izango den adibide bat sortzeko islapenak, errotazioak edo traslazioak erabiltzen dira. SMOTE algoritmoaren berezitasuna "ezaugarri espazioan"lan egitea da.

Adibide sintetikoak sortzeko k auzokide hurbiletan (*K Nearest Neighbors*) algoritmoan oinarritzen da. Algoritmoaren sasikodean ikusten dugun bezala auzokide hauek erabiltzen ditu interpolazio linear baten bitartez adibideak sortzeko.

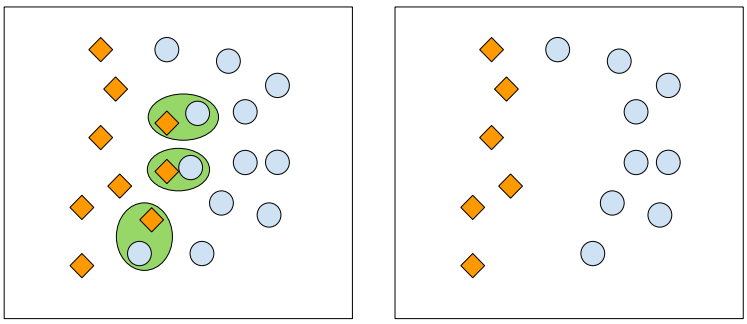
Algorithm 1: SMOTE

Input:
T klase gutxiengoaren adibideak
N sortu nahi den adibide kopuru ehunekoa
k auzokide hurbilenak
Output: $(N/100) * luzera(T)$ klase gutxiengoaren adibideak

```
1  N = (int)(N/100);
2  adibide_sintetikoak[];
3  indizea = 0;
4  for adibide in T do
5      auzokideak = auzokide_hurbilenak(adibide, T, k);
6      for i in N do
7          auzokidea = ausazko_auzokiea(auzokideak);
8          diferentzia = auzokidea − adibidea ;
9          adibide_sintetikoak[indizea] = adibidea + rand(0,1) · diferentzia;
10         indizea++;
11     end
12 end
13 return adibide_sintetikoak;
```

Tomek Links algoritmoa

Tomek (1976) proposatutako azpilangiketa algoritmo bat da. SMOTE algoritmoa bezala auzokide hurbiletan oinarritzen da adibideak ezabatzeko. Demagun sailkapen bitar baten aurrean gaudela, hau da, A eta B klaseak ditugula, izan bedi $x \in A$ eta $y \in B$ adibideak eta $d(x,y)$ x eta y adibideen "ezaugarri espazioko"distantzia. (x,y) *T-Link* bat izango dira



5.3 Irudia: Tomek Links algoritmoaren aplikazioaren aurretik eta ondorengo egoera.

baldin eta soilik baldin $\forall_z \in A \bigcup B - \{x,y\}$ izanda $d(x,y) < d(x,z)$ eta $d(x,y) < d(y,z)$. Beste modu batera esanda, x eta y bata bestearen auzokide hurbilenak direnean izango dira *T.Link*.

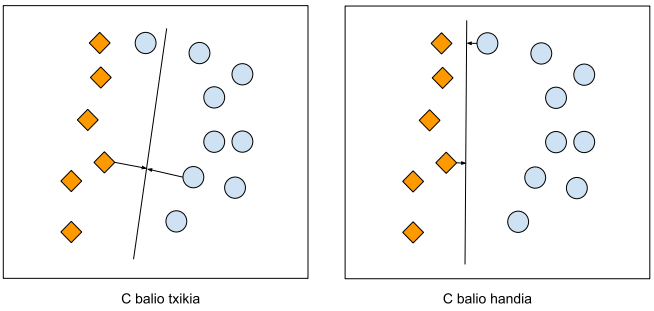
Algoritmo hau aipatutako *T-link*-ak ezabatzean datza. Bi modu aurkezten dira, adibide parea ezabatzea edo klase gehiengoaren adibidea soilik ezabatzea.

5.3. irudiari erreparatzen badiogu, ezkerrean *T-Link*ak berdez irudikatuta agertzen dira, aldiz, eskuinean, *T-link*ak osatzen zuten instantziak ezabatu ondoren lortutako egoera irudikatzen da.

5.1.5 Hiperparametroen optimizazioa

Ikasketa automatikoko algoritmoek bi motako parametroak dituzte: ikasketa momentuan doitutakoak eta aldez aurretik esleitutakoak. Atal honetan bigarren hauei buruz jardungo dugu.

SBM linearrak C hiperparametroa du. C-k adierazten du zenbaterainoko zigorra ezartzen zaion gaizki sailkatutako adibide bakoitzari. Zigorraren arabera izango da lortuko dugun hiperplanoaren itxura. C parametroari balio handiak emanez gero sailkatzailea saiatuko da ondo sailkaturiko entrenamenduko adibide kopurua igotzen; aldiz, balio txikiagoak



5.4 Irudia: C balio desberdinen araberako hiperplanoen adibideak.

ematen badizkiogu lortuko ditugun sostengu bektoreen eta hiperplanoaren arteko marjina handiagoa izango da.

C balio honek eragin handia dauka eredua egokitze prozesuan, hain zuzen ere, *C*ri balio handi bat emanez gero eredua gainegokitzera (*overfitting* ingelesez) eraman dezake, aldiz, balio txiki bat emanez gero azpiegokitzera (*underfitting* ingelesez) eraman dezake.

C parametroaren bilaketa egiteko algoritmo jale bat erabiltzea erabaki dugu (ikusi 2. algoritmoa) Algoritmo honen idea lauki-sare bilaketa behin eta berriz aplikatzea da, geroz eta tarte txikiago eta zehaztasun handiagoarekin.

5.2 Erlazio-erauzlearen esperimentazioa eta ebaluaketa

Atal honetan erlazio-erauzlearen sorkuntzarako egin ditugun esperimentazio eta erabakiak azalduko ditugu, baita egindako ebaluaketa eta lortutako emaitzak ere. Atal honetan zehar aipatuko ditugun zenbakizko emaitza guztiak 5.4. taulari egingo diote erreferentzia.

Algorithm 2: Lauki-sare bilaketa (*Grid search*)

Input:
iterazio_kopurua Algoritmoaren iterazio kopurua
C_0 Hasierako balioa
Output: C balio optimioa

$C = C_0 + \varepsilon$;
$d\_ = C_0$;
for *i* ***in*** *iterazio_kopurua* **do**
  $d = \frac{2}{5^i}$;
  $C\_balioak$ = range(from=$C - d\_$, to=$C + d\_ + d$, by=d) ;
  $C\_balioak = C\_balioak[C\_balioak > 0]$;
  $emaitzak$ = ebaluatu($C\_balioak$) ;
  $C = C\_balioak[\text{argmax}(emaitzak)]$;
end
return C;

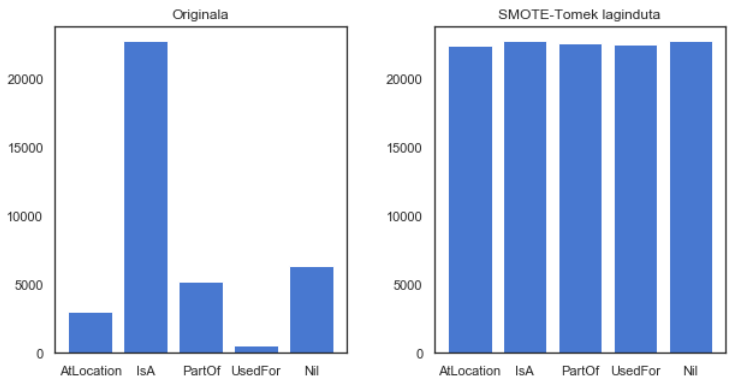
	Doitasuna	Estaldura	F1-Neurria
mintz	0.2116	0.2009	0.1574
+ opt	0.2121	0.2011	0.1581
+ lagin	0.2068	0.2019	0.1604
+ opt + lagin	0.2692	0.2202	0.1832
zhou	0.5605	0.4607	0.4718
+ opt	0.5648	0.4629	0.4752
+ lagin	0.5668	0.4664	0.4771
+ opt + lagin	**0.5332**	**0.5487**	**0.5329**

5.4 Taula: Doitasun, Estaldura eta F1-Neurria konfigurazio bakoitzarentzako makro batezbestekoa erabilita.

5.2.1 Ezaugarri moten konparaketa

Hasieran planteatutako ezaugarriak erabilita, hau da, Mintz et al. (2009)-en lanean oinarritutakoak, sistema ez da oso ondo moldatzen. Ebaluaketan lortutako emaitzek erakusten dute hasierako ezaugarrietan oinarrituta gure sistema ez dela gai izan *IsA* klasekoak ez diren adibideak ondo sailkatzeko. Hori dela eta, lortzen ditugun doitasuna, estaldura eta F1 neurriak oso baxuak dira: 0.21, 0.2 eta 0.15 urrenez-urren.

Zhou et al. (2005)-ren lanean oinarritutako ezaugarriekin ordea, lortutako emaitzak dexente igotzen dira (0.32 puntu F1 neurrian). Ezaugarri hauekin sailkatzailea hasten da



5.5 Irudia: Laginketa prozesua aplikatu aurretik eta ondorengo erlazio distribuzioa.

beste klaseetako adibideak ondo sailkatzen. Doitasun/estaldura kurbari erreparatzen badiogu (ikusi 5.7. irudia) kurba guztiz aldatzen da aurreko ezaugarriekin lortzen zenarekin konparatuz.

5.2.2 Entrenamendu datu-multzoaren laginketa

Distribuzio desorekak sor ditzakeen arazoak saihesteko lehen aipatutako **SMOTE** eta **TomekLinks** algoritmoak aplikatu ditugu entrenamenduko datu-multzoan adibide sintetikoak sortzeko. 5.5. irudian erakusten den bezala, laginketa algoritmoen ondorioz lortzen dugun distribuzio berria uniformea da.

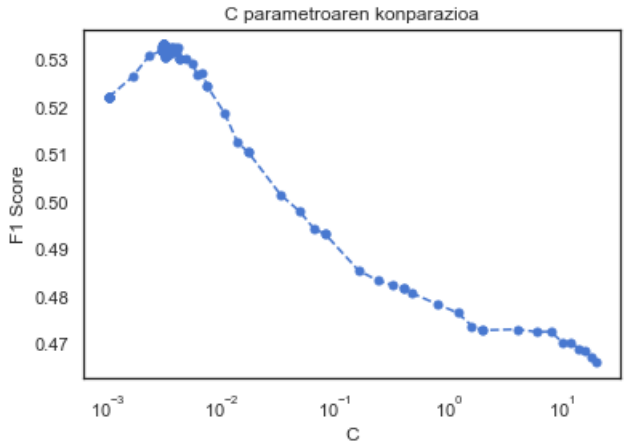
5.4. taulako emaitzei erreparatzen badiegu eragin oso handia eduki ez arren sistemak hobekuntza bat lortzen du kasu guztietan, batez ere, jatorrizko adibide gutxi zeuzkaten erlazioak sailkatzean. Beraz, ondoriozta dezakegu, laginketa metodoak erlazioen arteko desorekak sortzen dituen arazoetan laguntzen duela.

5.2.3 Hiperparametroen optimizazioa

Sailkatzaile baten parametroak bezain garrantzitsuak dira hiperparametroak, azken hauek lor dezakete portaera guztiz desberdinak sailkatzaileetan. Horregatik hiperparametroen balio aukeraketa egokia ezinbestekoa da.

Lehen aipatutako **lauki-sare bilaketa** (ikusi 2. sasikodea) algoritmoa aplikatu dugu **makro** F1 neurria optimizatu dezan garapen datu-multzoaren gainean.

Scikit-Learn liburutegiak eskaintzen duen LinearSVC algoritmoaren C hiperparametroaren defektuzko balioa 1 da. 5.6 irudian ikus dezakegun bezala, emaitzak nahiko hobetzen

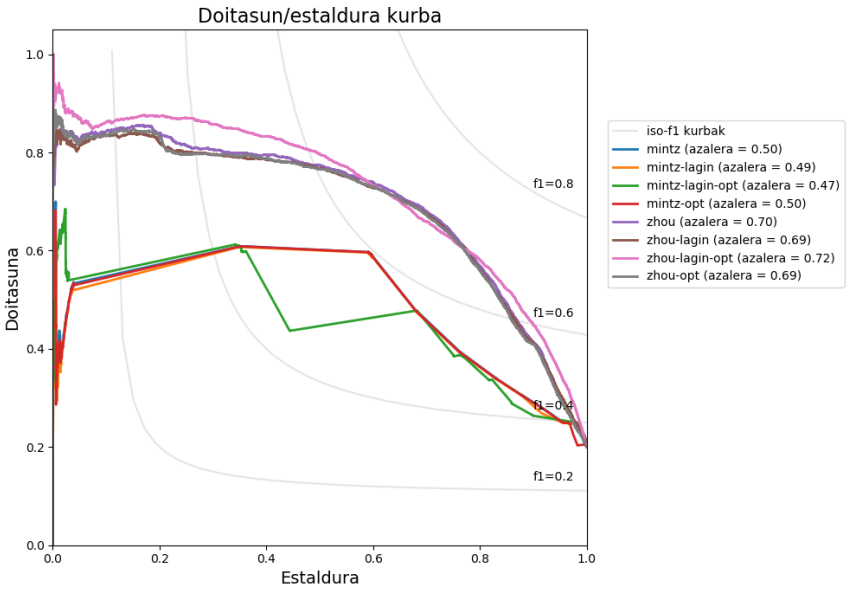


5.6 Irudia: Garapen datu-multzoaren gainean Zhou+opt+lagin konfigurazioan lauki-sare bilaketa-algoritmoaren eboluzioa.

dira, F1 neurriaren 0.05 puntu inguru, lehentsitako baliotik aldenduta. Gure kasuan esperimentu guztietan C balio optimoak oso txikiak izan dira, hain zuzen ere 10^{-3} gutxi gora-behera.

5.4. taulako emaitzei erreparatuta beti lortzen dugu hobekuntzaren bat, kasu batzuetan oso txikia izan arren.

5.2.4 Emaitzak eta errore-analisia



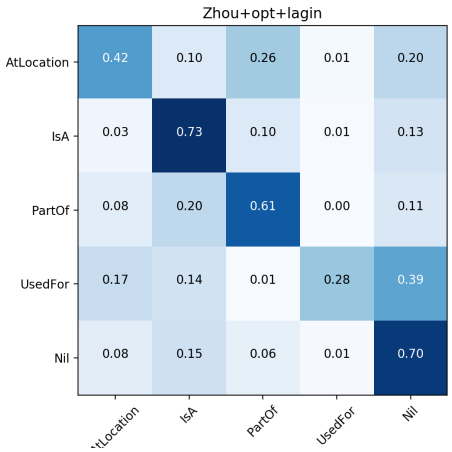
5.7 Irudia: Sistemen arteko doitasun/estaldura-kurbak mikro batezbestekoa erabilita.

Lortu ditugun emaitzarik onenak Zhou et al. (2005)-ren ezaugarriak erabilita laginketa eta optimizazioa eginda lortutakoak izan dira. Nabarmendu beharra dago bi hobekuntza nagusi egon direla prozesu osoan zehar; lehenengoa, ezaugarrien aldaketaren ondorioz gertatu da, eta, bigarrena, laginketa eta hiperparametro optimizazioen **konbinaketaren** eskutik etorri da.

5.4 taulan erakusten ditugu konfigurazio bakoitzaren **makro** ebaluazio metrikak. Ikus dezakegu bai Mintzen baita Zhouren ezaugarrien kasuan ere bakarrik laginketa edo bakarrik optimizazioa aplikatuta emaitzak ez direla asko aldatzen, aldiz, bi hauen konbinaketak hobekuntza nabariak ekartzen ditu.

Sistemen konparazio argiago bat edukitzeko 5.7 irudian doitasun/estaldura kurba azaltzen dugu. Irudian erabilitako neurriak aurreko taulan ez bezala **mikro** batezbestekoa erabiltzen dugu kalkulatzeko. Nabarmentzeko bi puntu dira nagusi, alde batetik, Mintzen ezaugarrien kasuan kurbaren azalera nahiko jaisten da laginketa eta optimizazioa aplikatzean, eta, beste aldetik, bai Mintzen baita Zhouren ezaugarrietan laginketa edo optimizazioa bakarrik aplikatuta kurbaren azalera pixka bat jaisten dela.

Errore-analisia



5.8 Irudia: Zhou+opt+lagin konfigurazioaren konfusio-matrizea.

Erlazio-erauzleak ze errore izan dituen argitzeko 5.8 irudian agertzen den konfusio-matrizean oinarrituko gara. Aipatutako konfusio matrizea emaitza onenak eman dituen konfigurazioari dagokio, hau da, Zhouren ezaugarriak, laginketa eta optimizazioa erabiltzen duen konfigurazioa. Konfigurazio horrek hurrengo erroreak erakusten ditu:

Alde batetik, laginketa aplikatu arren jatorrizko datu-multzoan adibide gehien zituzten klaseak sortutako konfusioa ez da guztiz desagertu. Hori dela eta, batez ere adibide gutxieneko klaseetan lortzen den estaldura nahiko baxua da.

Beste aldetik, *PartOf* eta *AtLocation* ez dira guztiz desberdinak, hau da, testuinguru oso antzekoak dituzte. Emaitzetan antzeman dezakegu nola *AtLocation*-eko ia adibide laurdena *PartOf* bezala sailkatzen dituen. Gure urruneko helburua erlazioetatik galderak sortzea dela gogoratuz, arazo honek ez du garrantzi handirik, hala ere, ebaluazio neurrietan nahiko eragina du.

Aurreko arazoarekin erlazionatuta dagoen beste bat aurkitzen dugu. Kasu honetan datu-multzo beraren arazoa da. Gerta daiteke argumentu bikote batek erlazio bat baino gehiago izatea. Fenomeno horri **etiketa anitzak** (*multi-label* ingelesez) deitzen zaio. Arazo hau jadanik ezaguna da erlazio-erauzketa atazan, lehenbiziko aipamena Hoffmann et al. (2011)-ek aurkezten dute soluzio bat ematen saiatuz. Gure ezagutza-baseko erlazio anitzeko adibide bezala <Vessels, Body> bikotea daukagu non *AtLocation* eta *PartOf* erlazioak esleituta dituen.

Azkenik, *UsedFor* klaseak aurkezten duen sailkapen arazoa. Gure sistema ez da erlazio hori zuzen sailkatzeko gai. Honen arrazoi nagusia klasearen adibide kopuru urria da. 4.9 taulan agertzen den bezala datu-multzoaren %1 besterik ez denez gure sistemak *Nil* bezala sailkatzen ditu adibide berrien gehiengoa.

6. KAPITULUA

Ondorioak eta etorkizuneko lanak

Atal honetan proiektuan zehar lortutako ondorioak azalduko ditugu, baita ondorio pertsonalak ere. Azkenik proiektuaren jarraipena izan daitezkeen etorkizuneko lanak azalduko ditugu.

6.1 Ondorioak

6.1.1 Lortutako emaitzak

Proiektuan planteatutako helburuak betetzea lortu ditugu, bai corpusaren sorkuntzan baita erlazio-erauzlearen inplementazioan ere.

Corpusaren sorkuntzari dagokionez, domeinuko terminologia erauzteko teknikak aztertu eta inplementatu dira, bai estatistikan oinarritutakoak baita datu-baseetan oinarritutakoak ere. Ezagutza-base baten azterketa egin da, eta, azkenik, corpusaren sorkuntza eta iragazketa garatu da. Laburbilduz, testu multzo batetik erlazio-erauzle sistema bat entrenatzeko behar den corpus baten sorkuntzarako garapen-lerro (*pipeline* ingelesez) bat aurkeztu da.

Erlazio-erauzlearen aldetik, jadanik existitzen diren sistemen azterketa bat egin da. Ikasketa automatikoaz baliatuz erlazio-erauzle bat inplementatu da, erlazio-erauzlearen ebaluazioa egiteko metrika objektiboak zehaztuz. Azkenik, hobekuntzak planteatu, inplementatu eta ebaluatu egin dira.

6.1.2 Ikasketa pertsonala

Formakuntzari dagokionez, oso esperientzia aberasgarria izan da, bai antolakuntza aldetik baita ezagutza aldetik ere. Alde batetik, orain arte egindako proiektuak baino handiagoa izan denez, denbora eta baliabideen kudeaketa proiektuaren atazen irismenean duten eragina ikasi dut. Bestetik, proiektuaren hasierako dokumentazio-fasetik eta proiektuaren garapenaren ondorioz jasotako ezagutza barneratu dut.

Proiektua ikerketa-talde batean garatu izana, taldearen funtzionamendua barrutik ikustea ahalbidetu dit. Bertan egiten diren lanen gaineko interesa piztu eta taldearen parte izateko nahia sortu dit. Ez bakarrik nik garatutako proiektuaren gaiaren inguruan, gaur egun dauden punta-puntako garapen zientifikoen jakinaren gainean egotea ahalbidetu dit.

Orain arte esperientzia handirik ez nuen arlo batean trebakuntza minimo bat lortu dut, Hizkuntzaren Prozesamenduan hain zuzen ere. Testua prozesatzeko tekniken berri izan dut, baita testuaren gaineko estatistikak ere. Hizkuntzekin trebetasun handia ez izan arren, arlo honek interesa piztu izana onartu beharra dut.

Azkenik, graduan zehar ikasitako kontzeptu askori etekina atera diedala esan dezaket. Horien artean, batez ere, Estatistikan eta Datu-Meatzaritzan ikasitako kontzeptuak izan dira nabarienak, bai corpusaren sorkuntzan baita erlazio-erauzlearen inplementazioan. Azken honetan ere *Machine Learning and Neural Networks* irakasgaian jasotako kontzeptu asko izan dira erabilgarri. Bestalde, Proiektu Kudeaketa izeneko ikasgaian ikasitakoa oso erabilgarria izan da proiektu honetako garapena ondo eraman ahal izateko. Azkenik, Sistema Eragileen Oinarriak irakasgaian Linux ingurunearen gaineko komando eta funtzionamendu orokorrak baliagarriak izan dira.

6.2 Etorkizuneko lanak

Mota honetako proiektuek ez dute bukaera finkorik. Hartutako erabaki bakoitzean adarkatze batetik bide bat aukeratzea besterik ez da, berriz, pauso bat eman ondoren beste adarkatze baten aurrean aurkitzeko. Modu horretan gauza berriak probatzeko eta jadanik dauzkagunak hobetzeko infinitu aukera aurkezten zaizkigu.

Proiektu batean baliabideak, denbora batez ere, mugatuak dira. Hori dela eta proiektuek irismen finko bat behar dute. Gure proiektuaren kasuan, sistema eraikitzera eta horri hobekuntza batzuk aplikatzera iritsi gara, hala ere, etorkizunean proiektuak izan dezakeen hainbat jarraipen-lerro bururatzen zaizkigu:

1. Gold Standarren sorkuntza.

   Orain arte erabili dugun testerako datu-multzoa zaratsua da, beste datu-multzoak bezala urruneko gainbegiraketaren bitartez sortu dugulako ere. Hori dela eta lortu ditugun emaitzek ez dute errealitatea guztiz erakusten. Gure sistemaren ebaluazioa modu egokiago batean egiteko, eskuz etiketaturiko testerako datu-multzo baten beharra dugu, *Gold Standard* bezala ere ezagutua.

2. Erlazio mota berrietara zabaltzea.

   Garatu dugun sistemak erakutsi du gai dela aukeratu ditugun lau erlazioak sailkatzeko. Gure azken helburua ahaztu barik, hau da, automatikoki galderak sortuko dituen sistema baten garapena, aberasgarria izan daiteke dibertsitatearen alde erlazio kopuru handiago bat edukitzea. Adibidez, pedagogikoki interesgarriak izan daitezkeen beste erlazio batzuk: *Causes*, *Entails* eta *CreatedBy* dira.

3. Ikasketa sakonean oinarritutako teknikak aplikatzea.

   Gaur egun indarrean dagoen ikasketa sakona aplikatzea izan daiteke eboluzio naturala. Hizkuntzaren prozesamenduaren arloan ez bakarrik beste askotan ere izugarrizko hobekuntza izan da ikasketa sakonaren aplikazioa. Ikusi dugu nola ezaugarrien gainean hobekuntza bat eginez gero hobekuntza hori emaitzetan islatzen dela; gure kasuan hobekuntza nabariena. Normala da pentsatzea guk erabilitako ezaugarriak baino hobeagoak egon daitezkeela. Hori dela eta, ikasketa sakonak eskaintzen duen ezaugarri-erauzketa automatiko horren bitartez emaitzak hobetzea espero dugu: 2.1.2. atalean aipatzen dugun Zeng et al. (2015) sistema, adibidez, gure problemara egokitzea eta probatzea.

4. Galdera-sortzaile sistema eraikitzea.

   Azkenean, proiektu honen motibazioa galderen sorkuntzan erabiltzeko sistema baten oinarri izatea da. Sistema hori eraikitzea eta bere funtzionamendua aztertzea izan daiteke aplikazio aldetik hurrengo pausoa.

Eranskinak

A. ERANSKINA

Proiektuaren Helburuen Dokumentua

Proiektu honen helburu nagusia galdera pedagogikoki esanguratsuagoak sortzeko erabiliko den erlazio-erauzle sistema baten garapena izango da. Garatuko dugun sistema ikasketa automatikoan oinarrituko da. Beste ikasketa automatikoko edozein sistema bezala egokitzeko corpus baten beharra du, gure ataza eta domeinurako ordea ez dugu corpus publikorik aurkitu, beraz, corpus honen sorkuntza ere proiektuaren helburu nagusitako bat izango da.

A.1 Hasierako erabakiak

Hasieratik Mintz et al. (2009)-ek aurkeztutako esperimentua gure domeinura hedatzea izan da gure ideia. Dokumentu-multzo bezala Wikipediako biologiako artikuluak aukeratu ditugu, modu aske eta errazean lortu daitekeen dokumentu-multzo eguneratu bat delako. Ezagutza-basearen aldetik ConcetpNet aukeratu dugu, arrazoi nagusia ConceptNetek jasotzen dituen erlazio multzoa gure domeinura ondo egokitzen delako da.

Proiektua garatzeko erabiliko den programazio-lengoaia *Python* izango da. Gaur egun programazio-zientifikoan, testu-prozesamenduan, ikasketa-automatikoan, software-garapenean, etab atazetan erabilia da. Eguneratutako liburutegi askoz osatuta dago, batez ere, ikasketa automatikoko liburutegiak dira gaur egun egiten dutenak *Python* indartsu.

A.2 LDE

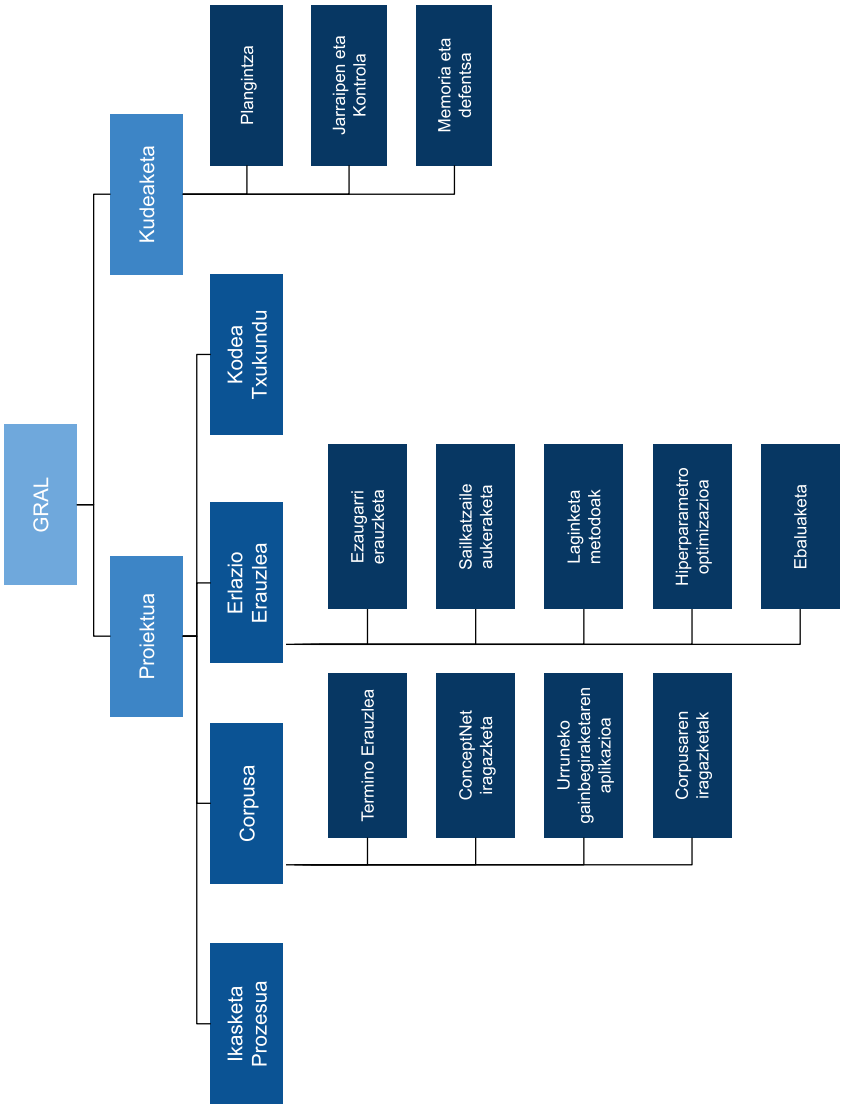
A.1. irudian agertzen da gure proiektuaren LDE-a (Lanaren Deskonposaketa Eredua). Irudian ikus daitekeen bezala alde batetik proiektuaren kudeaketa eraman dugu, eta, bestetik, proiektua bera. Bai proiektuaren, baita kudeaketaren lan-paketeak A.4. atalean daude zehatzago azalduta.

A.3 Gantt diagrama

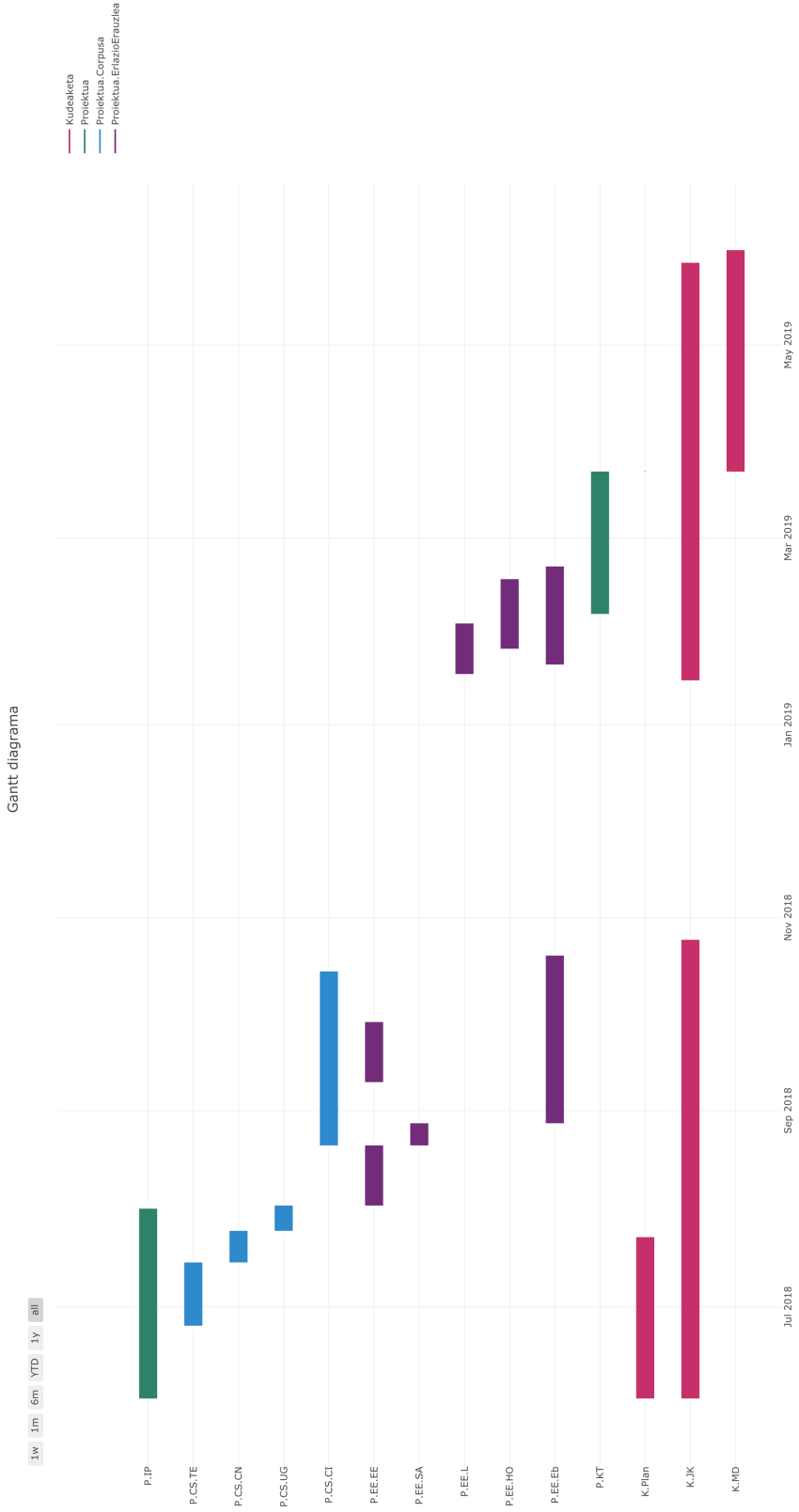
A.2. irudian ikus daiteke proiektuaren Gantt diagrama. Bertan ataza bakoitzaren hasiera eta bukaera adierazten da denboran zehar.

Gure proiektuaren garapena kronologikoki ulertzeko hiru fase bereiz daitezke. Alde batetik, ikasketa prozesua eta corpusaren sorkuntza osatzen duten fasea, corpusaren iragazketen eta lehendabiziko erlazio-erauzlearen garapenaren fasea, eta, azkenik, erlazio-erauzlearen hobekuntzaren fasea.

Proiektuan zehar etenaldi esanguratsu bat agertzen da azarotik urtarrilera bitartean. Etenaldi hori kurtsoaren azterketekin du zerikusia, horrek lehentasun bat duelako.



A.1 Irudia: Proiektuaren LDE diagrama.



A.2 Irudia: Proiektuaren Gantt diagrama

A.4 Lan-paketeak

Gradu Amaierako Lana, bi ataletan banatu da, alde batetik proiektua eta bestetik, kudeaketa.

Proiektuaren kasuan berriz lau ataza nagusi egon dira: ikasketa prozesu bat, corpusaren sorkuntza, erlazio-erauzlearen inplementazioa eta azkenik kodearen txukunketa bat. Ataza hauek hobeto zehazteko lan-paketeetan banatu dira.

Kudeaketaren aldetik hiru emangarri sortu dira: proiektuaren plangintza, proiektuaren gaineko jarraipen eta kontrola eta proiektuaren memoria eta defentsa.

A.4.1 Lan-paketeen deskribapen zehatza

1. Proiektua

   (a) **(P.IP) Ikasketa prozesua:** proiektuan beharrezkoak diren ezagutza minimoen ikasketa. Baita proiektua kokatzen den atazeko, erlazio-erauzketa hain zuzen ere, artearen egoerari buruzko ikerketa.

   (b) **(P.KT) Kodea txukundu:** proiektuan zehar sortutako kodea berantolatu eta txukundu ulergarria izateko eta bererabilpena sustatzeko.

   (c) Corpusaren sorkuntza

      i. **(P.CS.TE) Termino-erauzlea:** domeinua definituko duen hitzen zerrenda sortzea.

      ii. **(P.CS.CN) ConceptNet iragazketa:** gure proiekturako baliagarriak izango ez diren ezagutza-baseko sarrerak ezabatzea.

      iii. **(P.CS.UG) Urruneko gainbegiraketaren aplikazioa:** ezagutza-basea eta dokumentu multzoa edukita urruneko gainbegiraketa aplikatzea.

      iv. **(P.CS.CI) Corpusaren iragazketak:** zaratatsu izan daitezkeen adibideen identifikazioa eta iragazketa-prozesuak aplikatzea.

   (d) Erlazio-erauzlearen inplementazioa

      i. **(P.EE.EE) Ezaugarri-erauzketa:** erlazio-erauzketa atazean erabiltzen diren ezaugarriak aztertzea eta gure sistemarako erabiliko ditugun ezaugarriak aukeratzea/diseinatzea.

ii. **(P.EE.SA) Sailkatzaile aukeraketa:** ikasketa-automatikoan erabiltzen diren algoritmoen artean azterketa eta aukeraketa egitea.

iii. **(P.EE.L) Laginketa metodoen azterketa eta aplikazioa:** datu-multzo desorekatu baten aurrean agertzen diren arazoei aurre egiteko tekniken azterketa eta aplikazioa.

iv. **(P.EE.HO) Hiperparametroen optimizazioa:** sailkatzailearen hiperparametroak egokitzea.

v. **(P.EE.Eb) Ebaluaketa:** Sistemaren ebaluaketa kuantitatiboa.

2. Kudeaketa

   (a) **(P.K.Plan) Plangintza**

   (b) **(P.K.JK) Jarraipena eta kontrola**

   (c) **(P.K.MD) Memoria eta defentsa**

A.4.2 Lan-paketeen iraupena

Lan-paketeak			Estimatutako denbora (ordutan)
Proiektua	Ikasketa prozesua		70
	Corpusa	Termino-erauzlea	50
		ConceptNet irgazketa	30
		Urruneko gain. aplikazioa	30
		Corpusaren iragazketak	60
	Erlazio-erauzlea	Ezaugarri-erauzlea	40
		Sailkatzaile aukeraketa	20
		Laginketa metodoen azte. eta apl.	30
		Hiperparametroen opt.	20
		Ebaluaketa	40
	Kode txukunketa		40
Kudeaketa	Plangintza		30
	Jarraipen eta kontrola		50
	Memoria eta defentsa		90
GUZTIRA			600

A.1 Taula: Proiektuaren plagintzan estimatutako orduen desbiderapena

A.5 Emangarriak

Bi motatako emangarri sortu dira proiektu honetan:

- **Proiektuarekin lotutakoak:**
  - Erlazio-erauzle bat entrenatu ahal izateko automatikoki etiketaturiko corpus bat.
  - Aurreko corpusa sortzeko erabili den kodea.
  - Aipatutako corpusetik erlazio-erauzle baten inplementazioa garatzeko eta ebaluatzeko kodea, baita entrenatutako modeloa ere.
- **Proiektuaren kudeaketarekin lotutakoak:**
  - Proiektuaren memoria non proiektuaren deskribapenaz gain plangintza eta jarraipen eta kontroleko txostenak biltzen dituen.
  - Proiektuaren defentsarako gardenkiak.

A.6 Kalitatearen kudeaketa

Atal honetan proiektuaren kalitatea ebaluatzeko irizpideak zehaztu ditugu.

A.6.1 Kalitatearen plangintza

Proiektuaren kalitatea zehazteko bi kalitate maila definitu ditugu: Kalitate minimoa eta onargarria.

Kalitate maila minimoa

- **Corpusa:** urruneko gainbegiraketaren bitartez sortutako corpusa, zaratatsua izan arren sailkatzaile bat entrenatzeko erabilgarria dena.
- **Erlazio-erauzlea:** hasierako erlazio-erauzle sistema bat.
- **Kodea:** proiektuan lortu diren emaitzak errepikatu ahal izateko kodea.

- **Memoria:** Informatika Fakultateak proposatzen duen formatua jarraitzea, fakultateko web orrian agertzen den txantiloila erabiliz.
- **Gardenkiak:** Defentsarako ezarritako denboran oinarrituta gardenki kopurua erabiltzea.

Kalitate maila onargarria

- **Corpusa:** urruneko gainbegiraketaren ondorioz sortu den corpus zaratatsua iragazketa prozesu batetik pasatzea zarata desagerrarazteko asmoz.
- **Erlazio-erauzlea:** erlazio-erauzle sistemari hainbat hobekuntza aplikatzea ebaluaketa kuantitatiboan emaitzak hobetzen dituztenak.
- **Kodea:** Kodea ulergarri izatea eta ondo antolatuta egotea.

A.6.2 Kalitatearen kontrola

- **Corpusa:** corpusaren kalitatea benetan neurtzeko eskuzko ebaluazio bat behar da, baina, hori oso garestia denez, erlazio-erauzlea erabiliko da aplikatutako iragazketak emaitzak hobetzen dituzten edo ez esateko. Hala ere, adibide multzo txiki baten gaineko azterketa egingo da ere.
- **Erlazio-erauzlea:** sistemaren kalitatea neurtzeko 2.3. atalean aipatutako ebaluaziometrikak erabiliko dira ebaluaketa kuantitatiboa bat egiteko.
- **Memoria:** memoriaren kalitatea proiektuko zuzendariek ebaluatuko dute entregatu aurretik.
- **Gardenkiak:** gardenkien kalitatea proiektuko zuzendariek ebaluatuko dute entregatu aurretik.

A.7 Interesatuak

- IXA ikerketa taldeko ikerlariak.
- Hezkuntza arloko agenteak.
- Irakasleak.

A.8 Arriskuak eta prebentzioa

A.8.1 Arriskuak

- **Informazio-galera:** proiektuan garatutako kodea, corpusa edo bestelako informazio galtzeko arriskua.
- **Esperimentuak errepikagarriak ez izatea:** egindako esperimentu askotan ausazko elementuak agertzen dira. Ausazko elementu horiek egin dezakete hasiera batean lortutako emaitzak berriz ez lortzea.

A.8.2 Prebentzioa

- **Informazio-galera:** bai kodearen baita beste motatako informazioaren kopiak egin ditugu. Kodea garatzeko Git erabili dugu, eta, gure makinetan bakarrik gorde beharrean GitHub-era ere igo dugu. Beste informazioaren kasuan, corpusa adibidez, hainbat makinatan gorde dugu, alde batetik, IXA taldeko zerbitzarietan eta bestetik konputagailu pertsonaletan.
- **Esperimentuak errepikagarriak ez izatea:** egindako esperimentuak errepikagarriak izateko ausazko-hazi (*random seed* ingelesez) bat definitu dugu, horrela, ausazko elementuak erabiltzen dituzten algoritmoek beti emaitza berdinak itzultzeko.

A.9 Jarraipen eta Kontrola

Astero, bai proiektuaren ideiak eta esperimentuen emaitzak komentatzeko baita jarraipen eta kontrola egiteko ere bilerak egin dira. Bilera horietan egin da lan-pakete bakoitzaren jarraipena eta hurrengo lan-paketeen plangintzaren moldaketa.

Jarraipen eta kontrolari dagokionez planteatutako helburuak bete dira, baita A.5. atalean aipatutako emangarriak ere. Aipatutako emangarriak A.6. atalaren araberako bai kalitatea minimoa baita kalitate onargarria betetzen dute. Proiektuaren irismena bete dela esan dezakegu.

Proiektua IXA taldeko lan-poltsa bat bezala hasi zen, gero gradu amaierako lana bihurtze-

ko. Hori dela eta, 600 orduko proiektua definitu dugu, lan-poltsako 300 ordu gehi gradu amaierako proiektuko beste 300 ordu.

A.2. taulan ikus ditzakegu proiektuan suertatu diren orduen desbiderapenak hasierako plangintzarekiko. Aipatzeko bi desbiderapen esanguratsu daude, alde batetik, corpusaren sorkuntza osatzen duten lan-paketeek izan duten desbiderapen positiboa, eta, bestetik, erlazio-erauzlearen inplementazioa osatzen duten lan-paketeek izan duten desbiderapen negatiboa. Bi desbiderapenak kontuan harturik, estimatutako denbora eta denbora errealaren arteko diferentzia ez da oso handia izan.

A.9.1 Lan-orduen desbideraketa

Lan-paketeak			Estimatutako denbora (ordutan)	Denbora erreala (ordutan)	Desbiderapena
Proiektua	Ikasketa prozesua		70	70	0
	Corpusa	Termino-erauzlea	50	55	+5
		ConceptNet irgazketa	30	32	+2
		Urruneko gain. aplikazioa	30	25	-5
		Corpusaren iragazketak	60	63	+3
	Erlazio-erauzlea	Ezaugarri-erauzlea	40	42	+2
		Sailkatzaile aukeraketa	20	12	-8
		Laginketa metodoen azte. eta apl.	30	28	-2
		Hiperparametroen opt.	20	16	-4
		Ebaluaketa	40	42	+2
	Kode txukunketa		40	45	+5
Kudeaketa	Plangintza		30	23	-7
	Jarraipen eta kontrola		50	50	0
	Memoria eta defentsa		90	92	+2
GUZTIRA			600	595	-5

A.2 Taula: Proiektuaren plagintzan estimatutako orduen desbiderapena

Bibliografia

Angeli, G., Tibshirani, J., Wu, J., and Manning, C. (2014). Combining distant and partial supervision for relation extraction. pages 1556–1567.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2001). Latent dirichlet allocation. In *Advances in Neural Information Processing Systems 14 [Neural Information Processing Systems: Natural and Synthetic, NIPS 2001, December 3-8, 2001, Vancouver, British Columbia, Canada]*, pages 601–608.

Bowyer, K. W., Chawla, N. V., Hall, L. O., and Kegelmeyer, W. P. (2011). SMOTE: synthetic minority over-sampling technique. *CoRR*, abs/1106.1813.

Hoffmann, R., Zhang, C., Ling, X., Zettlemoyer, L., and Weld, D. S. (2011). Knowledge-based weak supervision for information extraction of overlapping relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 541–550, Stroudsburg, PA, USA. Association for Computational Linguistics.

Intxaurrondo, A., Surdeanu, M., López de Lacalle Lekuona, O., and Agirre Bengoa, E. (2013). Removing noisy mentions for distant supervision.

Li, S., Li, J., Song, T., Li, W., and Chang, B. (2013). A novel topic model for automatic term extraction. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '13, pages 885–888, New York, NY, USA. ACM.

M. Olney, A., Graesser, A., and Person, N. (2012). Question generation from concept maps. *Dialogue & Discourse*, 3.

Min, B., Li, X., Grishman, R., and Sun, A. (2011). New york university 2012 system for kbp slot filling. In *TAC*.

Mintz, M., Bills, S., Snow, R., and Jurafsky, D. (2009). Distant supervision for relation extraction without labeled data.

Mostow, J. and Chen, W. (2009). Generating instruction automatically for the reading strategy of self-questioning. In *Proceedings of the 2009 Conference on Artificial Intelligence in Education: Building Learning Systems That Care: From Knowledge Representation to Affective Modelling*, pages 465–472, Amsterdam, The Netherlands, The Netherlands. IOS Press.

Pershina, M., Min, B., Xu, W., and Grishman, R. (2014). Infusion of labeled data into distant supervision for relation extraction. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 732–738, Baltimore, Maryland. Association for Computational Linguistics.

Shen, W., Wang, J., and Han, J. (2015). Entity linking with a knowledge base: Issues, techniques, and solutions. *Knowledge and Data Engineering, IEEE Transactions on*, 27:443–460.

Smirnova, A. and Cudré-Mauroux, P. (2018). Relation extraction using distant supervision: A survey. *ACM Comput. Surv.*, 51(5):106:1–106:35.

Takamatsu, S., Sato, I., and Nakagawa, H. (2012). Reducing wrong labels in distant supervision for relation extraction. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1*, ACL '12, pages 721–729, Stroudsburg, PA, USA. Association for Computational Linguistics.

Tomek, I. (1976). Two modifications of cnn.

Yao, L., Haghighi, A., Riedel, S., and Mccallum, A. (2011). Structured relation discovery using generative models. pages 1456–1466.

Zeng, D., Liu, K., Chen, Y., and Zhao, J. (2015). Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1753–1762, Lisbon, Portugal. Association for Computational Linguistics.

Zhou, G., Jian, S., Zhang, J., and Zhang, M. (2005). Exploring various knowledge in relation extraction. *ACL-05 - 43rd Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*.