Ekaia, 2020, 37, 327-341
https://doi.org/10.1387/ekaia.20987

ekaia
ZIENTZIA eta TEKNOLOGIA ALDIZKARIA

ISSN 0214-9001 – eISSN 2444-3255

Euskarazko elkarrizketa sistema automatikoa sare neuronalen bidez

(A neural dialogue system in Basque)

Asier López Zorrilla, Mikel de Velasco Vázquez, Raquel Justo*

Elektrizitate eta Elektronika Saila, Euskal Herriko Unibertsitatea (UPV/EHU), Leioa

LABURPENA: Lan honetan sare neuronalen bidez euskaraz hitz egiten ikasten duen elkarrizketa sistema automatiko bat aurkezten dugu. Horretarako, Turingen testaren ideia era konputazionalean inplementatzen duten sare neuronal sortzaile aurkariak erabili ditugu. Normalean erabiltzen diren ingelesezko corpusak baino bi magnitude ordena txikiagoa den euskarazko corpus batekin halako sareak doitzea badagoela frogatzen dugu. Amaitzeko, euskararen morfologia kontuan hartzen duen aurreprozesamendua erabiltzea komenigarria dela erakusten dugu. Sare neuronaletan oinarrituta dagoen euskarazko lehen elkarrizketa sistema aurkezten dugu.

HITZ GAKOAK: elkarrizketa sistema automatikoak, sare neuronalak, sare neuronal sortzaile aurkariak, euskara.

ABSTRACT: *This work presents a neural dialogue system capable of learning Basque. To this end, we build upon generative adversarial networks which implement the idea of the Turing test. We demonstrate that training such a dialogue system with corpora two orders of magnitude smaller than usual English corpora is feasible. Finally, we also found that preprocessing the Basque language according to its morphology helps training these neural models. To the best of our knowledge, this is the first attempt to develop a neural dialogue system in Basque.*

KEYWORDS: dialogue systems, deep learning, generative adversarial networks, Basque language.

**** Harremanetan jartzeko / Corresponding author:*** Asier López Zorrilla. Speech Interactive Research Group, Elektrizitate eta Elektronika Saila, Zientzia eta Teknologia Fakultatea, Euskal Herriko Unibertsitatea (UPV/EHU), Sarriena Auzoa, zg, 48940, Leioa, Bizkaia, Euskal Herria. – asier.lopezz@ehu.eus – https://orcid.org/0000-0003-1739-1397.

Nola aipatu / How to cite: Eseberri, Itziar; López Zorrilla, Asier; De Velasco Vázquez, Mikel; Justo, Raquel (2020). «Euskarazko elkarrizketa sistema automatikoa sare neuronalen bidez»; *Ekaia*, 37, 2020, 327-341. (https://doi.org/10.1387/ekaia.20987).

Jasoa: 08 uztaila, 2019; Onartua: 29 urria, 2019.

ISSN 0214-9001 - eISSN 2444-3255 /

1. **SARRERA**

Elkarrizketa sistema automatikoek pertsona eta makinen arteko komunikazioa eta interakzioa ahalbidetzen dute, lengoaia naturalaren bidez. Adibide moduan azken urteotan hedatu diren laguntzaile birtualak aipa ditzakegu, hala nola Siri, Cortana, Google Assistant edo Alexa. Horiek normalean elkarrizketa sistema helburuduntzat hartzen dira, haien lana erabiltzailearen aginduak burutzea baita; esate baterako, dei bat egitea edo Interneten biharko eguraldiaren iragarpena bilatzea. Horietaz gain, busen ordutegiak eta lineak kontsultatzeko [1] eta jatetxeetan edo hoteletan erreserbak egiteko [2] balio duten sistemak helburudunak dira ere.

Beste alde batetik, aldez aurretik definitutako helbururik edo gairik ez duten elkarrizketa sistemak ere badira: eremu irekikoak. Sistema horietan erabiltzaileak eta makinak ez diote elkarri hitz egiten helburu espezifiko batekin; interakzioa bera naturala eta zentzuduna izatea da helburua. Horretarako, sistemak esaldi ahal bezain logiko, koherente eta informatzaileenekin erantzun behar dio erabiltzaileak esaten duenari. Beste modu batean esanda, sistemak era gizatiarrean hitz egin behar du. Lan honetan elkarrizketa sistema mota horietan zentratuko gara.

Era gizatiarrean hitz egitearen ideiarekin lotuta, Alan Turing matematikariak 1950. urtean bere test famatua aurkeztu zuen: Turingen testa [3]. Testaren ideia nagusia honakoa da: sistema automatiko bat kalitatezkoa edo adimenduna dela esateko, sistema horrek eta pertsona batek bereizezinak izan behar dute haiekin hitz egiteko orduan. Sistema batek halako propietatea betetzen duen egiaztatzeko, Turingek hainbat epaile zenbait makinarekin hitz egiten jartzea proposatu zuen, makina batzuen atzean sistema automatikoak eta besteen atzean pertsonak daudelarik. Egoera horretan epaileek proportzio handi[1] batean usteko balute sistema automatikoa pertsona bat dela, orduan sistema hori erabat adimenduna dela esan liteke.

Denbora pasatu ahala, Turingen testa gainditzearen ideiak gero eta ikerketa gehiago bultzatu zituen adimen artifizialaren arloan. Adibidez, 1966. urtean ELIZA programa [4] aurkeztu zuten MIT-eko ikertzaileek. Programaren funtsa hitz gakoak detektatzean eta horien arabera aurredefinitutako esaldi bat aukeratzean datza. Algoritmo hori sinplea izan arren, hainbat epailek pertsonatzat hartzea lortu zuen.

Hurrengo hamarkadetan ikerketek aurrera jarraitu zuten arren, benetan Turingen testa gainditzeko gai zen sistemarik ez zen lortu. 2011. urtean

[1] Eztabaida handia dago sistema batek Turingen testa gainditzeko behar duen portzentajearen inguruan. Erreferentzia gisa, 2011. urtean Indian Institute of Technology Guwahati institutuan ospatutako Turingen test batean, epaileek %63,3n pertsonak pertsona moduan sailkatu zituzten.

gauzak aldatu ziren, Turingen test batean inoiz lortutako emaitzarik onenak Cleverbot sistemak[2] lortu zituenean; berarekin hitz egin zuten 1.334 epaileetatik % 59,3-ak pertsonatzat hartu zuen. ELIZA-k ez bezala, Cleverbot-ek ez ditu aurredefinitutako esaldiak erabiltzen. Horren ordez urteetan zehar pertsonekin edukitako elkarrizketak erabiltzen ditu erantzuterako orduan. Hitz gutxitan esanda, esaldi bati erantzuteko Cleverbot-ek esaldi hori edo antzeko bat esan duenean zein erantzun jaso duen bilatzen du, eta erantzun horretatk abiatuta sortzen du bere erantzuna. Ideia hori interesagarria bada ere, konputazionalki nahiko garestia da, denbora zein memoriaren ikuspegitik, datu-base oso handi batean bilaketak egitea baitakar. Are gehiago, datu-basea zenbat eta handiagoa izan, orduan eta denbora eta memoria gehiago beharko du halako sistema batek erantzun bat sortzeko.

Eragozpen horiek saihesteko, baita adimen artifizialaren beste arloetan izan duten emaitzengatik ere, azken urteetan sare neuronalak elkarrizketa sistemak eraikitzeko teknologia nagusia bilakatu dira. Sare neuronalak datuetatik eredu konputazional konplexuak lortzeko balio duten paradigma konputazional bat dira, bereziki eraginkorra datuen kantitatea oso handia denean. Ulertzekoa da, beraz, arloko autore gehienek ingelesez dauden datu-baseekin lan egitea, horiek izanda baitira handienak, eta, hortaz, sare neuronalek hobeto funtzionatuko dutelako. Baina zer gertatzen da baliabide gutxiagoko hizkuntzekin? Ba al dago sare neuronaletan oinarrituriko elkarrizketa sistema automatikoak eraikitzerik euskaraz?

Lan honetan erakusten dugu baietz, badagoela. Normalean erabiltzen diren datu-baseak baino bi magnitude ordena txikiagoko datu-baseak erabiliz modu koherente eta zentzudunean euskaraz hitz egiten duen elkarrizketa sistema automatikoa aurkezten dugu.

2. ARLOKO EGOERA ETA IKERKETAREN HELBURUAK

Sare neuronalen bidezko eremu irekiko elkarrizketa sistemak itzulpen automatikorako erabiltzen diren sareetan oinarritzen dira, hots, sekuentziatik sekuentziarako sareetan [5, 6] (*Sequence to sequence networks* ingelesez). Sare neuronal horiek luzera arbitrarioko bektore segida bat har dezakete sarrera moduan, eta era berean beste luzera arbitrario bateko segida bat sortu. Hala, transdukzio problemak ebazten saiatzeko baliagarriak dira. Itzulpen automatikoaren kasuan, sarreran hizkuntza batean idatzitako esaldia hartuko du sareak, eta irteeran esaldi hori beste hizkuntza batean sortu. Bestalde, elkarrizketa sistemak eraikitzerako orduan, sarrera erabiltzaileak esandako hitzen segida izango da, eta irteera sistemaren erantzunari dagozkion hitzen sekuentzia.

[2] https://www.cleverbot.com/, azken bisita 2019ko uztailaren 1ean.

Sare horiek entrenatzeko edo haien parametroak doitzeko, ikasketa metodo gainbegiratuak erabili ohi dira, aipatutako sarrera-irteera bikoteez osaturiko corpusen bat erabiliz. Adibidez, lan honetan filmen azpitituluak erabiliko ditugu corpusa eratzeko: sarrera bakoitza aktore batek esandako esaldi bat izango da, eta dagokion irteera beste aktore batek emandako erantzuna.

Metodologia hori erabiliz emaitza interesgarriak lortu ahal diren arren, askotan horrela entrenatutako sareek informaziorik gabeko erantzun orokorrak sortzeko joera dute, hala nola «*I don't know* » edo «*I'm sorry* » [7, 8]. [9] lanean adierazten den moduan, ikasketa metodo gainbegiratuek irteera bakarra esleitzen diote sarrera bakoitzari, baina horrek ez ditu elkarrizketen propietateak behar bezala jasotzen. Izatez, hitz egiten dugunean, norbaitek esan duenari erantzuteko hamaika esaldi erabili ahalko genituzke, guztiak onargarriak. Horrela, esaldi askoren erantzuna izan daitezkeen esaldi generikoak probabilitate handiarekin sortzen ditu sareak.

Arazo hori konpontzeko, ikasketa gainbegiratuaren ordez sare sortzaile aurkariak (*Generative adversarial networks* ingelesez) [10] erabiliko ditugu lan honetan. Sare sortzaile aurkariek Turingen testaren ideia era konputazionalean aplikatzea ahalbidetzen dute. Kasu honetan, erantzunak sortzen dituen sareari (sare sortzailea hemendik aurrera) ez zaio adieraziko zer irteera dagokion sarrera bakoitzari. Horren ordez, beste sare batek, sare diskriminatzaileak, ebaluatuko ditu sare sortzaileak emandako erantzunak, zein punturaino diren gizatiarrak esanez, Turingen testaren epaile batek egingo lukeen modu berean. Sare sortzailearen helburua sare diskriminatzaileak berari emandako ebaluazioa ahal bezainbeste hobetzea izango da. Sare diskriminatzailearena, aldiz, pertsonek sortutako eta sare sortzaileak sortutako esaldien artean bereiztea izango da. Modu horretan, bi sareak iteratiboki entrenatuko dira; sortzailea saiatuko da diskriminatzaileak hura pertsonatzat har dezan, diskriminatzaileak sortzailearen eta pertsonen artean bereizten ikasten duen bitartean.

Halako optimizazio prozesua egitea, dena den, ez da sinplea, sareak entrenatzeko normalean erabiltzen diren gradienteetan oinarritutako optimizazio metodoak ez baitira zuzenean aplikagarriak. Xehetasunetan sartu gabe, sare diskriminatzailearen irteera ez da diferentziagarria sare sortzailearen parametroekiko, sortzaileak sortutako hitzak diskretuak dira eta [11]. Errefortzu bidezko ikasketa erabili daiteke gradienteetan oinarritutako metodoen ordez [12, 13], baina horrek entrenamenduaren konbergentzia zaildu dezake [14]. Beste aukera bat *straight-through Gumbel-softmax* [15, 16] zenbateslearen bidez gradientearen hurbilketa bat egitea da, [17] eta [18] laneetan erakusten duten moduan. Azkenik, lan honetako autoreek guztiz diferentziagarria den sare sortzaile aurkari bat aurkeztu berri dute [19], hitzen errepresentazio bektorial hurbilduak erabiltzen dituena, ondoren azalduko dugun moduan.

Testuinguru horretan, lan honen ekarpenak hiru dira: alde batetik, [19] lanean proposatutako sare sortzaile aurkaria balioztatzen dugu, ingelesez ez ezik euskaraz ere eraginkorra dela frogatuz; bigarrenik, modu koherente eta zentzudunean hitz egiten duen sare neuronaletan oinarritutako elkarrizketa sistema automatikoa euskaraz eraikitzea badagoela frogatzen dugu; eta, amaitzeko, lematizazio prozesu baten bidez corpusaren tamaina txikiagoagatik sortutako desabantailak nola leundu daitezkeen erakusten dugu.

3. atalean, proposatutako elkarrizketa sistema osatzen duten sareen egituraren deskribapena emango dugu. Hurrena, 4. atalean sare horien parametroak doitzeko egingo saiakuntzak eta lortutako emaitzak erakutsiko ditugu, eta, 7. atalean, ondorio batzuk eta etorkizunerako planteatzen den norabidea aipatuko ditugu.

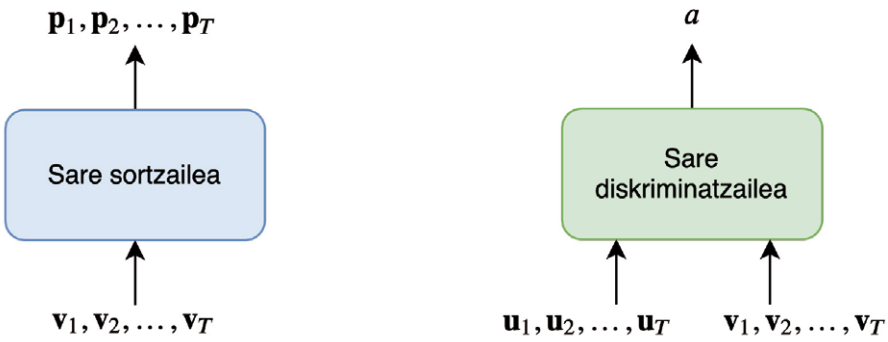
3. SARE SORTZAILE ETA DISKRIMINATZAILEA

Esan bezala, elkarrizketa sistema eraikitzeko sare sortzaile aurkariak erabili ditugu. Beraz, bi sare entrenatu behar dira elkarrizketa sistema lortzeko: sare sortzailea, sarrera mezu baten aurrean erantzuna sortzen duena; eta sare diskriminatzailea, sarrera mezu baten aurrean emandako erantzun bat ebaluatzen duena.

Sare sortzailea sekuentziatik sekuentziarako sare bat da, *long short-term memory* edo LSTM [20] kodetzaile eta deskodetzaile errekurrente independenteekin [5] eta arreta modulu batekin [21, 22]. Sare horrek T luzera arbitrarioko hitzen errepresentazio bektorialen [23] segida bat hartuko du sarrera moduan: $\mathbf{v} = \mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_T$. Sarrera hori prozesatu ostean, irteera moduan beste τ luzera arbitrarioko segida bat bueltatuko du, elementu bakoitza sareak sor ditzakeen hitz guztien arteko probabilitate-banaketa izanik: $\mathbf{p} = \mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_\tau$.

Bestalde, sare diskriminatzailea beste bi kodetzaile errekurrentez osaturik dago, guztiz konektatutako geruza batzuez jarraituak, [24] lanean azaltzen den antzera. Kodetzaile bakoitzak esaldi bat hartzen du sarrera moduan. Batek **v**, erabiltzailearen mezua, prozesatuko du, eta besteak **u**, erantzuna. Sistemaren irteera 0 eta 1-en arteko zenbaki erreal bat, a, izango da, erabiltzailearen mezuari emandako erantzuna gizatiarra zer punturaino den adierazten duena. Irteera zenbat eta baxuagoa, orduan eta gizatiarragoa izango da erantzuna, sarearen irizpidearen arabera. Bi sarrerak, berriz ere, hitzen errepresentazio bektorialen moduan hartuko ditu sareak.

Bi sareen sarrerak eta irteerak 1. irudian erakusten ditugu. Gainontzeko xehetasunak [19] lanean aurki daitezke.



1. irudia. Sare sortzailearen eta diskriminatzailearen sarrerak eta irteerak.

4. IKASKETA ALGORITMOA

Bi sareak entrenatzeko, hiru optimizazio prozesu era iteratiboan egingo ditugu. Lehen aipatu dugun moduan, alde batetik, sare sortzailea entrenatuko dugu diskriminatzaileak hura pertsonatzat har dezan, hau da, diskriminatzailearen irteera minimizatzeko. Bigarrenik, diskriminatzailea entrenatuko dugu sare sortzaileak sortutako erantzunak eta corpusetik hartutako erantzunak desberdintzeko. Amaitzeko, [12] lanean proposatzen den legez, sare sortzailearen parametroak ikasketa metodo gainbegiratuen bidez doituko ditugu ere noizean behin, prozedura orokorraren konbergentzia bermatzeko.

Optimizazio prozesu horiek definitzeko, horietako bakoitzean gradienteetan oinarritutako optimizazio metodoekin minimizatuko ditugun galera-funtzioak zehaztuko ditugu.

4.1. Sare sortzailearen parametroen egiantz handieneko zenbatespena

Sare sortzailearen parametroak ikasketa gainbegiratuaren bidez doituko ditugu egiantz handieneko zenbatespen baten bidez. Hau da, corpuseko sarrera-irteera bikote bakoitzarentzat, sareak sarrera prozesatzean irteera desiratua sortzeko duen probabilitatea maximizatuko dugu. 1. ekuazioan agertzen den galera-funtzioa erabiliko dugu.

$$L_{EH} = \frac{1}{|C|} \sum_{\mathbf{v},s \in C} \frac{1}{|s|} \sum_{t=1}^{|s|} -\log \mathbf{p}_t[s_t] \tag{1}$$

non C $\mathbf{v}$ sarrerek eta s irteera desiratuek osatutako corpusa den, s_t irteera desiratuaren t-garren hitzari dagokion indizea den, eta $\mathbf{p}_t$ $[s_t]$ sareak t-ga-

rren denbora unean *st* hitzari esleitutako probabilitatea den. Sarearen irteera, **p**, **v** sarreraren menpe dago noski, baina mendekotasun hori ez dugu esplizituki adierazi notazioa ez korapilatzeko.

4.2. **Sare diskriminatzailearen galera-funtzioa**

Sare diskriminatzailearen entrenamendua egiteko lehenik eta behin corpus berri bat sortu beharko dugu, C_D, C corpusetik abiatuz eta sare sortzailea erabiliz. Diskriminatzaileak pertsonek emandako eta sare sortzaileak sortutako erantzunak desberdintzen ikasi behar duenez, bi eratako laginak behar ditu bereb artean diskriminatu ahal izateko. Horretarako, bi motatako hirukoteez osatuko dugu C_D corpusa. Hirukote bakoitza erabiltzaileak bidalitako mezu batez, erantzun batez eta 0 edo 1 izan daitekeen etiketa batez osatuta egongo da. Lehenengo motako hirukoteek gizakiek emandako erantzunak edukiko dituzte, eta, beraz, etiketa 0 izango da. Hirukote horiek lortzeko C corpuseko bikoteak erabili ditugu zuzenean. Bestalde, bigarren motako hirukoteek sare sortzaileak sortutako erantzunak edukiko ditu, eta, ondorioz, etiketaren balioa 1 izango da. Hirukote horiek eratzeko, C corpusetik hartu dira erabiltzailearen mezuak, gero horiek sare sortzaileari pasatu sarrera moduan, eta sarearen irteera erabili erantzun moduan. C_D eraiki ondoren, entropia gurutzatuko galera-funtzioa erabili dugu sare diskriminatzailearen parametroak doitzeko:

$$L_D = \frac{1}{|C_D|} \sum_{\mathbf{v},\mathbf{u},l \in C_D} -\left[l \cdot \log a + (1-l) \cdot \log(1-a)\right], \tag{2}$$

non **v** erabiltzailearen mezuaren hitzen errepresentazio bektorialen segida den, **u** erantzunarena, l erantzuna pertsona batena edo sare sortzailearena den adierazten duen eskalarra, eta a diskriminatzailearen irteera. Berriro ere, a-k **v**-rekiko eta **u**-rekiko duen mendekotasuna ez dugu esplizituki adierazi.

4.3. **Sare sortzailearen galera-funtzio aurkaria**

Azkenik, sare sortzailea diskriminatzailearen irteera minimizatzeko galera-funtzioa definitzea erraza da, diskriminatzailearen irteera bera baita, 3. ekuazioan ageri den bezala.

$$L_S = \frac{1}{|C_S|} \sum_{\mathbf{v} \in C_S} a\,, \tag{3}$$

non C_S corpusa C corpusean dauden sarrera mezuez osatuta dagoen, **v** horietako bakoitza izanik. a diskriminatzailearen irteera da.

3. ekuazioko galera-funtzioa gradienteetan oinarritutako optimizazio metodoekin minimizatu ahal izateko, *a* sare sortzailearen parametroekiko diferentziagarria izan behar du. Sare sortzaileak **v** sarrera **p** irteeran era guztiz diferentzialean transformatzen du. Era berean, sare diskriminatzaileak bere bi sarrerak, **v** eta **u**, era guztiz diferentzialean transformatzen ditu *a* irteeran. Hortaz, diferentziagarritasuna ez galtzeko, **p** **u**-n transformatu behar da transformazio diferentziagarri baten bidez. **p**-ko elementu bakoitza, hots, $\mathbf{p}_t$, sareak esan ditzakeen hitz guztien arteko probabilitate-banaketa bat da. Normalean $\mathbf{p}_t$-ko maximoaren argumentua hartuko genuke sareak *t*-garren denbora unean esan duen hitzat. Baina argmax operazioa ez da deribagarria.

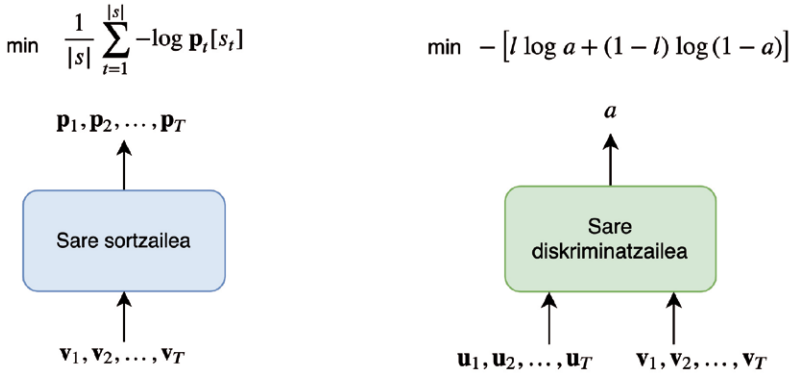
Arazo horri irtenbidea emateko, [19] laneko prozedura berdina erabiltzen dugu lan honetan. $\mathbf{p}_t$-ri dagokion errepresentazio bektoriala, $\mathbf{u}_t$, lortzeko, $\mathbf{p}_t$-ko *k* elementurik handienak hartzen ditugu, *top-k* operazio baten bidez. Horrela, elementu horien $\tilde{\mathbf{p}}_t$ balioak eta $\mathbf{k}_t$ indizeak lortzen ditugu. Jarraian, $\tilde{\mathbf{p}}_t$ normalizatzen dugu *softmax* normalizazio batekin, eta $\tilde{\mathbf{p}}_t t$ lortu. Azkenik, $\mathbf{u}_t$ kalkula dezakegu $\mathbf{k}_t$ indizeei dagozkien hitzen errepresentazio bektorialen arteko batazbesteko aritmetiko haztatua eginez, pisuak $\hat{\mathbf{p}}_t$ izanik.

4.4. **Goi mailako optimizazio algoritmoaren deskribapena**

Erabiliko diren hiru galera-funtzioak deskribatu ondoren, horiek iteratiboki minimizatzeko prozedura zehaztuko dugu. Hasteko, sare sortzailearen parametroak ez ditugu ausaz hasieratuko. Horren ordez, hainbat iteraziotan zehar doituko ditugu hasieran, 1. ekuazioko egiantz handieneko galera-funtzioa minimizatuz. Behin sare sortzaileak kalitate onargarriko esaldiak sortuz gero, C_D corpusa bere erantzunekin eta *C*-ko pertsonen erantzunekin hasieratuko dugu, eta sare diskriminatzailea lehenengo aldiz entrenatuko dugu.

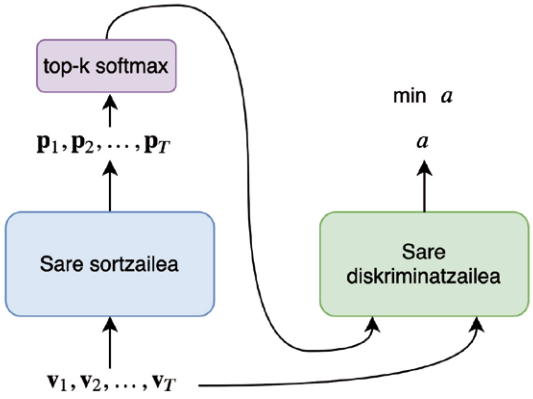
Ondoren, algoritmoaren begizta nagusia hasten da. Horretan, sare sortzailea eta diskriminatzailea iteratiboki entrenatzen dira. Sare sortzailea entrenatzeko galera-funtzio aurkaria (3. ekuazioa) eta egiantz handieneko galera-funtzioak (1. ekuazioa) txandakatzen dira. Prozedura osoan zehar, sare sortzailearen entrenamendu prozesu bakoitza amaitu ostean, hainbat sarrera ausaz aukeratzen dira, *C*-tik eta sare sortzaileak sortutako erantzunak C_D-ra gehitzen dira, eta diskriminatzailea entrenatzen da hainbat iteraziotan zehar (2. ekuazioa). Prozeduraren konbergentzia bermatzeko, diskriminatzailea entrenatzerakoan probabilitate handiagoarekin hartzen dira C_D-n sartutako erantzun berriagoak.

2. irudian algoritmoan egindako hiru optimizazio problemen adierazpen grafikoa erakusten dugu. Kasu bakoitzean optimizatzen den galera funtzioa agertzen da, sarrera-irteera bikote bakarrarentzat.



(a) Sare sortzailearen parametroen egiantz handieneko zenbatespena.

(b) Sare diskriminatzailearen entrenamendua.



(d) Sare sortzailearen entrenamendu aurkaria.

2. irudia. Entrenamendu algoritmoan egiten diren optimizazio prozesuen laburpena.

5. EUSKARAREN AURREPROZESAMENDUA ETA LEMATIZAZIOA

Orain, euskaraz dagoen corpusa nola aurreprozesatu dugun deskribatuko dugu. Esan dugunez, euskarazko corpus batekin entrenatuko ditugu sareak. Ingelesa ez bezala, euskara hizkuntza eranskaria da egitura morfologikoaren aldetik. Hau da, euskarak monema independenteak elkartuz sortzen ditu hitzak. Horrela, askotan euskaraz hitz batekin esan daitekeena ingelesez hainbat hitz erabiliz adierazi behar da. Adibidez, ingelesezko «*to the cinema*» euskaraz «zinemara» itzuliko litzateke, edo «*because of the*

baby» «haurrarengatik». Sareen ikuspegitik hitz bakoitza token independente bat denez, sareak ez ditu ikusten euskaraz gertatzen diren hitzen arteko erlazioak, eta horrek euskararen prozesamendu automatikoa zailtzen du. Hasiera batean, behintzat, sarearentzat «haurrarengatik» eta «haurraren» hitzak «haurrarengatik» eta «daitezke» bezain ezberdinak dira.

Honek hitzen errepresentazio numerikoa zailtzen du bi sareen sarreran, baita sare sortzailearen irteeran ere. Sareen sarreretan, hitzen egituran arreta jartzen duten errepresentazio bektorialak erabiliko ditugu hitzen arteko erlazio horiek sortzeko, *Fastext* [25] hain zuzen ere. Dena den, irteeran ezin da arazoa horrela konpondu, sare sortzaileak hitzen arteko probabilitate-banaketa bat sortzen duelako. Horri irteera ematen saiatzeko, hitzen lexemak kasu marketatik eta postposizioetatik banatzea proposatzen dugu. Zehazki, izen, izenordain, adjektibo eta determinanteak bananduko ditugu lan honetan.

Hitzen lexema eta kategoria gramatikala topatzeko, [26] lanean aurkeztutako kode irekiko lematizatzailea erabiliko dugu. Izen, izenordain, adjektibo edo determinante baten lexema eta postposizioa banatuko diren ala ez erabakitzeko, baldintza sinple bat erabiliko da. Halako hitz baten bukaera postposizio baten berdina balitz, orduan hitzak lexematan eta postposiziotan bananduko dugu. Adibidez, «zeruko» hitza «zeru» lexeman eta «-ko» postposizioan banatuko genuke. Hogeita bi postposizio eta kasu marka hartu genituen kontuan: «-ri», «-ei», «-rekin», «-ekin», «-ren», «-en», «-n», «-tik», «-dik», «-rik», «-ra», «-tara», «-rengana», «-engana», «-rantz», «-raino», «-z», «-rako», «-ko», «-entzat», «-tzat» eta «-gatik».

Lematizazioaz gain, izen propioak <izen> tokenera bihurtuko ditugu, normalean pertsonen izenak baitira, eta, beraz, funtzio berdina dutelako esaldietan. Era berean, zenbakiak <zenbaki> tokenera bihurtuko dira [26] laneko lematizatzailea erabili genuen bi ataza hauetarako ere.

6. **SAIAKUNTZAK ETA EMAITZAK**

Orain arte azaldutako sareak, ikasketa algoritmoa eta euskararen aurreprozesamendua balioztatzeko, OpenSubtitles [27] corpusaren euskarazko bertsioarekin entrenatu dugu deskribatutako elkarrizketa sistema. Corpus horretatik milioi bat sarrera-irteera bikote atera ditugu, ingelesezko bertsioan baino 420 aldiz gutxiago. Corpusa 5. atalean azaldutako metodologiarekin aurreprozesatu dugu; ondorioz hitz desberdinen kopurua berrehun milatik ehun milara jaitsi da. Normalean egiten den bezala, hitz horietako azpimultzo bat baino ez dugu kontuan hartuko saiakuntzetarako: maiztasun handieneko 15.000 hitzak. Gainontzekoak corpusetik kendu dira. Aurreprozesamenduaren efektua erakusteko, corpus aurreprozesatua zein aurreprozesatu gabearekin entrenatu ditugu sareak.

Kasu bietan, dena den, hiper-parametro berdinak erabiliko ditugu sareen arkitekturan eta baita ikasketa algoritmoan. Hiper-parametro horietatik inportanteenak jarraian aipatzen ditugu. Sare errekurrente guztiak, hau da, sare sortzailearen kodetzailea, deskodetzailea, eta sare diskriminatzailearen bi kodetzaileak, bi LSTM geruzaz osatuta daude. Sare sortzailearen geruzek 1.028 zelda dituzte, eta diskriminatzailearenak 128. Adam optimizazio metodoa [28] erabiliko dugu 4. ataleko hiru galera-funtzioak minimizatzeko, 512 tamainako *batch*-ak erabiliz. Hitzen errepresentazio bektorialak *Fastext* metodologiarekin hasieratuko dira, eta entrenamenduan zehar optimizatuko dira. Sare sortzailea 50.000 iteraziotan zehar entrenatuko dugu, ikasketa begizta hasi baino lehen. Hori 500 aldiz errepikatu dugu ondoren. Iterazio bakoitzean sare sortzailea zein diskriminatzailea 40 iterazioetan zehar entrenatuko da.

[29] lanean adierazten den moduan, ebaluazio automatikoak ez dira komenigarriak elkarrizketa sistemen kalitatea neurtzeko, normalean ez baitago korrelazio nabaririk horien eta pertsonek egindako ebaluazioen artean. Hortaz, entrenatutako sistemen funtzionamendua erakusteko, zenbait sarrerako mezuen aurrean emandako erantzuna erakusten dugu emaitza modura. Erreferentzia gisa, metodologia berdinarekin baina OpenSubtitles corpusaren ingelesezko bertsioarekin entrenatutako sareak emandako erantzunak ere erakusten ditugu[3]. Erantzun guztiak 1. taulan ageri dira.

7. ONDORIOAK ETA ETORKIZUNERAKO PLANTEATZEN DEN NORABIDEA

1. taulan ikusi daitekeen moduan, sare neuronal sortzaile aurkarien bidez euskaraz era nahiko koherente eta zentzudunean hitz egiten duen elkarrizketa sistema automatikoa lortu dugu. Ingelesarekin konparatuz euskaraz dauden baliabideen tamaina askoz txikiagoa izan arren, sare neuronalen bidezko metodologiak erabiltzea badagoela frogatu dugu. Horretarako, euskararen morfologia kontuan hartzea inportantea dela ere erakutsi dugu. Izen, izenordain, adjektibo edo determinanteak lexematan eta postposiziotan banatzea komenigarria da, sareak hala era eraginkorrago batean prozesatzen baitu lengoaia. 1. taulari begira, aurreprozesu horrekin sareak esaldi konplexuagoak sortzeko joera duela esan dezakegu.

Gainera, lan honekin proposatu berri dugun [19] eta testuarekin era guztiz diferentzialean lan egin dezakeen sare sortzaile aurkarien arkitektura baliozkotzen dugu, eta elkarrizketa sistema automatikoak euskaraz eraikitzeko aproposa dela egiaztatu.

---

[3] Ingelesez corpusaren tamaina handiagoa denez, sareak ere handiagoak dira, eta iterazio gehiagotan zehar entrenatu dugu. Zehaztasunak [19] erreferentzian ematen dira.

1. taula. Hainbat mezuren aurrean elkarrizketa sistemek emandako erantzunak. **A** ingelesez entrenatutako sistema da, **B** euskaraz aurreprozesamenduarekin entrenatutakoa eta **D** aurreprozesamendurik gabekoa

Sarrera mezua	Sistemaren erantzuna
A: hi !	hello , <person>.
B: kaixo !	kaixo , <izen> naiz , <izen>-ren semea naiz .
D: kaixo !	kaixo , kaixo.
A: I love you	you know , I think maybe we could talk ...
B: maite zaitut	nik ere maite zaitut .
D: maite zaitut	maite zaitut
A: who is she ?	she 's my girlfriend , <person> , and ...
B: nor da neska hori ?	<izen> da neska .
D: nor da neska hori ?	nor da neska ?
A: what's the color of the sky ?	it 's a beautiful blue sky
B: zein da zeruko kolorea ?	zeru urdin iluna da .
D: zein da zeruko kolorea ?	zein da zure kolore faboritoa ?
A: do n't worry	I do n't want to hear about it.
B: ez kezkatu	ez dut ulertzen zergatik egin duen hau zutzat, <izen> ?
D: ez kezkatu	ez dut ahaztuko
A: could you come with us there ?	I 'll come with you .
B: toki horitara lagunduko diguzu ?	bai , horixe uste dut nik ere .
D: toki horretara lagunduko diguzu ?	jakina baietz
A: I forgot it .	I did n't know you were coming .
B: ahaztu egin zait .	<izen> .
D: ahaztu egin zait .	zer ?
A: can you see anything ?	I see you in the mirror .
B: zerbait ikusten ?	ez , ez
D: zerbait ikusten ?	ez .
A: it was a plane crash .	I was in the hospital
B: hegazkin istripu bat izan zen .	<izen> larri zaurituta zegoen eta bere etxera buel-tatu zen
D: hegazkin istripu bat izan zen .	sute bat izan zen .
A: did he wake up again ?	no he did not, but he did n't sleep .
B: jaiki al da berriz ?	zer esan nahi du horrek ?
D: jaiki al da berriz ?	astelehenean atera behar duzu .
A: yesterday I saw you in the mountains	and now I see you
B: atzo menditik ikusi zintudan	ikusi zintudan <zenbaki> aldin .
D: atzo menditik ikusi zintudan	zer ?
A: see you	see you later , <person> .
B: agur	agur , aita .
D: agur	agur , ene erregea

Dena den, lan honetan aurkeztutako metodologia eta ideiak asko garatu behar dira benetan pertsona baten moduan euskaraz hitz egiten duen sistema lortzeko. Izatez, ingelesez ere oraindik urrun gaude halako sistemak sortzetik. Oraingoz baliabide handiagoko eta txikiagoko lengoaiekin sortutako sistemak parekatzea da gure hurrengo helburua. 1. taulan ageri den moduan, ingelesez entrenatutako sare sortzaile aurkariak era zentzudunagoan eta gizatiarragoan hitz egiten du euskarazko sistemarekin konparatuta.

Diferentzia horiek murrizteko, ezagutzaren transferentzia (*transfer learning* ingelesez) egiteko teknikak erabiltzeko asmoa daukagu. Ezagutzaren transferentziaren ideia nagusia da corpus handiagoekin baina eginkizun ezberdin baterako entrenatutako ereduak eredu berriak sortzeko erabiltzea. Kasu honetan, beraz, ingelesez sortutako sarea euskarazko sistema hobetzeko erabiltzea izango da gure helburua.

Amaitzeko, etorkizunean [30] lanean proposatutako eta *byte pair encoding* edo BPE izeneko aurreprozesamenduarekin saiakuntzak egingo ditugu, eta guk proposatutako lematizazioarekin konparatu. BPE-a tokenizatzaile edo hitz-banatzaile estatistiko eta automatikoa da, guk proposatutako banaketak ez ezik, printzipioz beste banaketa zentzudun batzuk ere egiteko gai dena.

8. **BIBLIOGRAFIA**

[1] OLASO J.M. eta TORRES M.I. 2017. «User experience evaluation of a conversational bus information system in spanish». *8th IEEE International Conference on Cognitive Infocommunications*.

[2] BORDES, A. eta WESTON, J. 2016. «Learning end-to-end goal-oriented dialog». *CoRR abs/1605.07683*.

[3] TURING A.M. 1950. «Computing machinery and intelligence». *Mind*, LIX, 433-460.

[4] WEIZENBAUM, J. 1966. «ELIZA - a computer program for the study of natural language communication between man and machine». *Communications of the ACM*, 9, 36-45.

[5] SUTSKEVER I., VINYALS O. eta LE Q.V. 2014. «Sequence to sequence learning with neural networks». *Advances in neural information processing systems*, 3104-3112.

[6] CHO K., MERRIE¨ NBOER B., C¸ AGLAR G., BAHDANAU D., BOUGARES F., SCHWENK H., eta BENGIO Y. 2014. «Learning phrase representations using RNN encoder-decoder for statistical machine translation». *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing EMNLP*, 1724-1734.

[7] SORDONI A., GALLEY M., AULI M., BROCKETT C., JI Y., MITCHELL M., NIE J., GAO J. eta DOLLAN B. 2015. «A neural network approach to

context-sensitive generation of conversational responses». *Human Language Technologies: The 2015 Annual Conference of the North American Chapter of the ACL*, 196-205.

[8] SERBAN I.V., SORDONI A., BENGIO Y., COURVILLE A. eta PINEAU J. 2016. «Building end-to-end dialogue systems using generative hierarchical neural network models». *Thirtieth AAAI Conference on Artificial Intelligence*.

[9] TUAN Y. eta LEE H. 2019. «Improving conditional sequence generative adversarial networks by stepwise evaluation». *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

[10] GOODFELLOW I., POUGET-ABADIE J., MIRZA M., XU B., WARDE-FARLEY D., OZAIR S., COURVILLE A. eta BENGIO Y. 2014. «Generative adversarial nets». *Advances in neural information processing systems*, 2672-2680.

[11] YU L., ZHANG W., WANG J. eta YU Y. 2017. «SeqGAN: Sequence generative adversarial nets with policy gradient». *AAAI*, 2852-2858.

[12] LI J., MONROE W., SHI T., JEAN S., RITTER A. eta JURAFSKY D. 2017. «Adversarial learning for neural dialogue generation». *arXiv preprint arXiv:1710.06547*

[13] HORI T., WANG W., KOJI Y., HORI C., HARSHAM B. eta HERSHERY J.R. 2019. «Adversarial training and decoding strategies for end-to-end neural conversation models». *Computer Speech & Language*, 54, 122-139.

[14] SUTTON R.S. eta BARTO A.G. 1998. «Introduction to reinforcement learning». *MIT press Cambridge*.

[15] BENGIO Y., LÉONARD N. eta COURVILLE A. 2013. «Estimating or propagating gradients through stochastic neurons for conditional computation». *arXiv preprint arXiv:1308.3432*.

[16] JANG E., GU S. eta POOLE B. 2016. «Categorical reparameterization with gumbel- softmax». *arXiv preprint arXiv:1611.01144*.

[17] LU J., KANNA A., YANG J., PARIKH D. eta BATRA D. 2017. «Best of both worlds: Transferring knowledge from discriminative learning to a generative visual dialog model». *Advances in Neural Information Processing Systems*, 314-324.

[18] SHETTY R., ROHRBACH M., HENDRICKS L.A., FRITZ M. eta SCHIELE B. 2017. «Speaking the same language: Matching machine to human captions by adversarial training». *Proceedings of the IEEE International Conference on Computer Vision*.

[19] LÓPEZ ZORRILLA A., DEVELASCO VÁZQUEZ M. eta TORRES M.I. 2019. «A differentiable generative adversarial network for open domain dialogue». *Tenth International Workshop on Spoken Dialogue Systems Technology (IWSDS)*.

[20] HOCHREITER S. eta SCHMIDHUBER J. 1997. «Long short-term memory». *Neural computation*, 9, 1735-1780.

[21] BAHDANAU D., CHO K. eta BENGIO Y. 2016. «Neural machine translation by jointly learning to align and translate». *CoRR abs/1409.0473*.
[22] LUONG M., PHAM H. eta MANNING C.D. 2015. «Effective approaches to attention- based neural machine translation». *arXiv preprint arXiv:1508.04025*.
[23] MIKOLOV T., CHEN K., CORRADO G.S. eta DEAN J. 2013. «Efficient estimation of word representations in vector space». *CoRR abs/1301.3781*.
[24] KANNAN A. eta VINYALS O. 2017. «Adversarial evaluation of dialogue models». *arXiv preprint arXiv:1701.08198*.
[25] BOJANOWSKI P., GRAVE E., JOULIN A. eta MIKOLOV T. 2016. «Enriching word vectors with subword information». *arXiv preprint arXiv:1607.04606*.
[26] RODRIGO A., BERMUDEZ J. eta RIGAU G. 2014. «Ixa pipeline: Efficient and ready to use multilingual NLP tools». *LERC*, 3823-3828.
[27] LISON P. eta TIEDEMANN J. 2016. «Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles». *Proceedings of the 8th International Conference on Language Resources and Evaluation*.
[28] KINGMA D.P. eta BA J. 2014. «Adam: A method for stochastic optimization». *arXiv preprint arXiv:1412.6980*.
[29] LIU C., LOWE R., SERBAN I., NOSEWORTHY M., CHARLIN L. eta PINEAU J. 2016. «How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation». *EMNLP*.
[30] SENNRICH, R., HADDOW, B., BIRCH, A. 2016. «Neural Machine Translation of Rare Words with Subword Units». *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, 1715-1725.