eman ta zabal zazu

EUSKAL HERRIKO UNIBERTSITATEA

University of the Basque Country

PhD dissertation

Towards general attribute controllability in NLP models

Aitor Ormazabal

2023

eman ta zabal zazu

EUSKAL HERRIKO UNIBERTSITATEA

University of the Basque Country

Towards general attribute controllability in NLP models

Aitor Ormazabal Oregik Eneko Agirre Bengoa eta Mikel Artetxe Zurutuzaren zuzendaritzapean eginiko tesi-txostena, Euskal Herriko Unibertsitatearen Doktore titulua eskuratzeko aurkeztua.

Donostia, 2023ko Abendua.

Esker ona

Eskerrik asko...

... Mikel eta Enekori, maila pertsonal zein profesionalean zuzendari ezin hobeak izateagatik. Eskerrik asko ere Aitor, Gorka eta Manexi, proiektu desberdinetan zuekin elkarlanean aritzea plazer bat izan da.

... IXA eta ixakideei, eskainitako bizipen eta barre guztiengatik. Doodleak betetzeko zuen atzetik ibiltzea faltan botako dut.

... Holger, Loic, and the great team at FAIR Paris.

... Dani and Che, for an awesome summer internship. I look forward to continuing working together.

... Familia eta lagunei, beti hor egoteagatik.

This work was supported by a FPU grant from MECD.

Abstract

Advances in deep learning methodology and computing infrastructure have yielded impressive results in the field of Natural Language Processing (NLP) in recent years. However, the core paradigm followed by deep learning methods has not changed much in the past decade. Deep learning models derive their behavior entirely from their training data and learning objective, and often do not offer any mechanisms to control or steer their outputs. Thus, if one wants to control a certain aspect of the model's output, one needs to gather training data that explicitly demonstrates the desired attribute, which is not always feasible or practical.

The goal of this thesis is to address this issue by designing methods to control diverse attributes of output of NLP, beyond the existing paradigm of simply gathering more training data and re-training the model.

In the first section of the thesis we focus on unsupervised methods that allow for controllability when training data that exemplifies the desired attribute is not available. We develop three methods for different model architectures, in accordance with the evolution of the field during the development of this thesis.

First, we propose a method to control the alignment of static word embeddings during training without any bilingual supervision, and apply it to train state-of-the-art —at the time of publication— unsupervised bilingual word embeddings.

Second, we leverage the information bottleneck technique, together with an adversarial training setup, to control the information content in the encoded representation of a sequence-to-sequence model, and apply it to develop a paraphrase system from bilingual corpora. We prove mathematically that our method alleviates issues inherent to the popular round-trip translation baseline for paraphrasing, while offering a natural way to control the tradeoff between diversity and fidelity in the paraphrases.

Third, we explore the use of control codes to train a meter- and rhyme-controllable language model, and develop PoeLM, an unsupervised poetry generation model for

Basque and Spanish. We show for the first time that control codes can be used for the control of fine-grained and strict attributes such as meter and rhyme patterns, and evaluate our method through both automatic metrics and human evaluation. We find that human evaluators often rate equally or prefer short poems generated by PoeLM to those written by layman human volunteers.

Having developed several unsupervised methods for different architectures and attributes, the second part of this thesis focuses on a general method for arbitrary adaptation of language models. Particularly, we focus on the scenario where one wants to adapt a language model when access to the internals of the model is not possible. This scenario has become particularly relevant in recent years, where, due to both the extreme scale of modern language models and the proliferation of black-box models hidden behind APIs, one often cannot simply fine-tune the model's weights for adaptation. To this end, we present CombLM, a method for black-box language model adaptation, that first trains a fine-tuned small "expert" model on the target task or domain, and then combines it with the black-box model at the probability level through a learned combination, to obtain an adapted model. Our approach allows us to leverage the deep knowledge of existing large models, while retaining the flexibility to adapt them to new domains and tasks. We show the effectiveness of our approach for adaptation to several domains and one downstream machine translation task.

Note for non-Basque speaking readers

This dissertation is structured as a collection of articles. The introductory chapter is in Basque, whereas the conclusions and the articles themselves are in English. Non-Basque speaking readers are recommended to first read the Conclusions chapter to get an overview of the main contributions made at this thesis, followed by the papers in the appendix in their recommended reading order.

Laburpena

Ikasketa sakoneko metodologian eta konputazio-azpiegituretan egindako aurrerapenek emaitza ikusgarriak ekarri dituzte Hizkuntzaren Prozesamenduaren arloan azken urteotan. Hala ere, metodo hauek jarraitzen duten oinarrizko paradigma ez da asko aldatu azken hamarkadan. Ikasketa sakoneko ereduek beren portaera datu multzotik eta ikasketa-helburutik eratortzen dute oso-osorik, eta askotan ez dute inolako mekanismorik eskaintzen beren irteerak kontrolatzeko. Beraz, ereduaren irteeraren atributu jakin bat kontrolatu nahi bada, nahi den atributua esplizituki adierazten duten datuak bildu behar dira, eta hori ez da beti bideragarria edo praktikoa.

Tesi honen helburua arazo honi aurre egitea da, datu gehiago bildu eta eredua berriro entrenatu behar izan gabe ereduen irteerako hainbat atributu kontrolatzea ahalbidetzen duten metodoak diseinatuz.

Tesiaren lehen atalean, gainbegiratu gabeko metodoetan zentratuko gara, nahi den atributua adierazten duten datuak eskuragarri ez daudenean erabili daitezkeenak. Hiru metodo garatzen ditugu arkitektura desberdinetarako.

Lehenik, entrenamenduan zehar hitz-bektore estatikoen lerrokaketa kontrolatzeko metodo bat proposatzen dugu, inolako gainbegiratze elebidunik gabe funtzionatzen duena, eta artearen egoerako—argitalpen unean—gainbegiratu gabeko hitz-bektore elebidunak entrenatzeko erabiltzen dugu.

Bigarrenik, informazio-mugatzearen teknika baliatzen dugu, ikasketa antagonikoarekin batera, kodetzaile-deskodetzaile eredu baten adierazpen kodetuaren informazio-edukia kontrolatzeko, eta corpus elebidunetatik abiatuta parafrasi-sistema bat garatzeko aplikatzen dugu. Matematikoki frogatzen dugu gure metodoak pibote bidezko itzulpen automatikoan oinarritutako metodoen berezko arazoak arintzen dituela, eta parafrasietan aniztasunaren eta fideltasunaren arteko trukea kontrolatzeko modu naturala eskaintzen duela.

Hirugarrenik, kontrol-kodeen erabilera aztertzen dugu, sortutako testuaren metrika

eta errima kontrolatzea ahalbidetzen duen hizkuntza-eredu bat entrenatzeko. Teknika hau baliatzen dugu PoeLM garatzeko, euskarazko eta gaztelaniazko poesia-sorkuntza eredu bat. Lehen aldiz erakusten dugu kontrol kodeak erabil daitezkeela mota honetako atributu xeheak zehazki kontrolatzeko, eta gure metodoa ebaluazio automatikoen eta giza ebaluazioaren bidez ebaluatzen dugu. Giza ebaluatzaileek PoeLMek sortutako poema laburrak giza boluntario ez-adituek idatzitakoekin alderatzean sarritan berdin baloratzen dituztela edo nahiago dituztela frogatzen dugu.

Arkitektura eta atributu desberdinetarako gainbegiratu gabeko hainbat metodo garatu ondoren, tesi honen bigarren zatia hizkuntza-ereduen egokitzerako metodo orokor baten garapenean zentratzen da. Bereziki, kutxa-beltz ereduetan zentratzen gara, non ereduaren barne-funtzionamendua atzitu edo eraldatzea ezinezkoa den. Eszenatoki hau bereziki garrantzitsua bihurtu da azken urteotan, non, egungo ereduen eskala erraldoia dela eta, edo APIen atzean ezkutatzen diren ereduen hedapena dela eta, sarritan ereduaren parametroak edo barne-funtzionamendua ezin den eraldatu. Horretarako, CombLM aurkezten dugu, kutxa-beltz hizkuntza-ereduak egokitzeko metodo bat. Lehenik eta behin eredu "aditu" txiki bat entrenatzen dugu helburuko ataza edo domeinuan, eta ondoren kutxa-beltz ereduarekin konbinatzen dugu probabilitate mailan, ikasitako konbinaketa-funtzio baten bidez. Gure hurbilpenak eredu handien ezagutza orokorra baliatzeko aukera ematen digu, domeinu eta zeregin berrietara egokitzeko malgutasuna mantenduz. Eredu handi bat hainbat domeinu berrietara eta itzulpen automatiko ataza batera egokituz gure hurbilpenaren eraginkortasuna frogatzen dugu.

Contents

1 Sarrera

Sarrera hau hurrengo moduan egituratua dago: 1.1 atalean tesiaren aurkezpena egiten da, eta gaiaren zergatia motibatzen da. 1.2 atalean tesiko lan-lerro eta helburu desberdinak azaltzen dira, eta 1.3 atalean lan-lerro bakoitzean garatutako lanak eta lortutako emaitzen laburpena azaltzen da. 1.4 atalean tesiko esparru teorikoaren eta erabilitako metodologien oinarriak deskribatzen dira, eta 1.5 atalean erlazionatutako lan nagusiak azaltzen dira.

1.1 Motibazioa

Azken hamarkadan adimen artifizialeko arloaren bilakaera ikusgarria izan da. Ikasketa automatikoan —partikularki, neurona-sareetan oinarritutako ikasketa sakonean— egindako aurrerapenei esker, adimen artifiziala ikerkuntza akademikoaren esparrutik atera, eta gizarteko maila guztietan erabiltzen hasi den erreminta bat izatera igaro da. Besteak beste, ikasketa automatiko ereduak daude egungo itzulpen automatiko, ahots-sintesi, ahots-transkripzio, argazkilaritza digital, eta aurpegi-antzemate sistemen atzean. Baina agian gaur egun entzute handieneko adimen artifizial sistemak ChatGPT bezalako asistente-birtual orokorrak dira, testu-interfaze baten bitartez erabiltzaileari edozein atazarekin laguntza eskaintzen diotenak. Mota honetako asistenteek hizkuntzaren prozesamenduan ibilbide luzea duten arren, 60. hamarkadan arloaren hastapenetik (Weizenbaum, 1966), hamarkada honetan ChatGPTren agerpena arte ez dira praktikan erabilgarriak izan.

Arrakasta hau ahalbidetu duten aurrerapen ugari egon diren arren —bai ikerkuntza aldetik, bai ingeniaritza aldetik, eta bai konputazio azpiegitura aldetik— ikasketa sakoneko ereduak entrenatzeko modua funtsean ez da aldatu: datu multzo handiak biltzen dira, eta ondoren eredua optimizatzen da ikasketa-helburu bat maximizatzeko. Adibidez, itzulpen automatiko eredu bat entrenatzean, neurona-sare batek esaldi bat jaso, eta itzulpen posible desberdinen probabilitateak itzuliko ditu. Orduan, ikasketa-helburuak neurona-sareari benetako itzulpenaren probabilitatea maximizatzen irakatsiko dio. Modu honetan, datu multzo eta ikasketa-helburu desberdinekin neurona-sareek ikasi dezakete testua itzultzen, ahotsa transkribatzen, eta beste ataza asko betetzen.

Paradigma orokor hau izugarri arrakastatsua izan den arren, ez du entrenatutako ereduak sortzen dituen irteeren gainean kontrol xeherik ahalbidetzen. Adibidez, demagun itzulpen eredu bat entrenatzen ari garela. Horretarako, milioika esaldi-itzulpen pare bilduko genituzke, eta ondoren neurona-sare bat optimizatu esaldi bat emanda haren itzulpena aurresaten ikas dezan. Bildutako datu multzoa —eta neurona-sarea— nahiko handiak baldin badira, sistemak kalitatezko itzulpenak sortzen ikasiko du. Baina, demagun gure eredua erabiltzen hastean, konturatzen garela hizkera oso informala erabiltzeko joera duela; hitz bat itzultzeko hainbat aukera daudenean, sarritan informalena aukeratzen duela, alegia. Zer egin genezake portaera hau desegokia baldin bada, adibidez sistema hau testuinguru profesional batean erabiltzeko asmoa baldin badugu? Ohiko ikasketa sakoneko metodoek ez dute eskaintzen horrelako **atributuak kontrolatzeko** —kasu honetan, formaltasun maila— modu erraz bat. Paradigma klasikoan, formaltasun maila kontrolatzeko modu bakarra, nahi dugun formaltasuna duten datu-masa handiak bildu eta entrenatzeko erabiltzea izango zen.

Ikus dezagun beste adibide bat. Demagun gure helburua bertsoak sortzeko gai den sistema bat entrenatzea dela. Gaur egungo arkitektura neuronalak kalitatezko testu naturala sortzeko gai dira, datu multzo eta ikasketa-helburu egokiekin entrenatzen badira. Beraz, bertso sortzailea entrenatzeko bertso kopuru handi bat bilduko dugu, eta hizkuntza-eredu bat entrenatzeko erabiliko dugu.[1] Hizkuntza-eredu honek entrenamenduan ikusitakoaren antzekoa den testuak sortuko ditu, hau da, bertsoak. Baina, ereduak bertso mota zehatz bat sortzea nahi badugu, adibidez, *zortziko txikia*, ziurtatu beharko dugu entrenamendu testu guztiek egitura metriko hori jarraituko dutela: 8 lerro, 7/6 silabako lerroak tartekatzen, eta 6 silabako lerroak errimatzen. Baina, zer gertatzen da ondoren *zortziko handiak* —7/6 silabako lerroak erabili ordez 10/8 silabako lerroak erabiltzen dituena– sortu nahi baditugu? Beste datu multzo guztiz desberdin bat bildu beharko dugu, mota honetako bertsoz osatua. Bistan da estrategia hau ez dela eraginkorra: bertso mota berri bakoitzeko, datu multzo berri bat beharko dugu, eta sistema berriz entrenatu beharko dugu. Bertso sortzaile orokor bat entrenatu ahal izateko, benetan nahiko genukeena hizkuntza-ereduak sortutako testuaren metrika eta errima **kontrolatu** ahal izatea izango litzake. Hau sortutako testuari gehitu nahi zaion murriztapen baten beste adibide bat da. Baina, hizkuntza-ereduak entrenatzeko ohizko estrategia jarraituta, **ezin dira horrelako murriztapenak gehitu**, eta ondorioz berariazko datu multzo bereziak beharko lirateke metrika bakoitza ereduari irakasteko.

Aurreko bi adibideek hizkuntzaren prozesamendu ereduak —edo edozein adimen

[1] Hizkuntza-ereduak hurrengo hitz aurresate (HHA) izeneko ikasketa-helburuarekin entrenatzen diren ereduak dira: testu bat emanda, ereduak testu horren ondoren etorriko den hurrengo hitzaren gaineko probabilitate banaketa aurresaten ikasten du. Modu honetan entrenatzearen ondorioz, eredu hauek gai dira testu naturala sortzeko. Aurrerago azalduko ditugu hizkuntza-ereduak.

artifizial sistema— entrenatzean sarritan aurkitzen den arazo bat azpimarratzen dute: Batzuetan sistema bat entrenatu nahi dugu ataza nagusi bat betetzeko —aurreko kasuetan, itzulpen automatikoa edo testu sorkuntza— baina sistemaren irteeren atributu jakin batzuk kontrolatu nahi ditugu —testuaren formaltasun maila, edo egitura metrikoa— atributu hauek esplizituki adierazten dituzten datu multzoak izan gabe. Baina, arestian aipatutako ikasketa sakoneko paradigma orokorrak ez du mota honetako atributuen kontrola lortzeko modu errazik eskaintzen. Beraz, nola arindu dezakegu arazo hau, kontrolagarriak diren sistemak eraikitzeko?

Galdera hau erantzutea da tesi honen helburua. Zehazki, testua darabilten hizkuntzaren prozesamenduko sistemetan zentratuko gara, eta hauen kontrolagarritasuna lortzeko teknika desberdinak aztertzea dugu helburu. Ikerketa-lerro honen garrantzia justifikatzeko bi arrazoi nagusi nabarmendu ditzakegu:

Interes zientifikoa. Arestian aipatu bezala, ikasketa sakonean entrenatutako ereduaren portaera soilik datu multzoaren eta ikasketa-helburuaren araberakoa izango da. Ondorioz, paradigma klasikoan sistema baten irteeraren atributu bat kontrolatzeko aukera nagusia datu multzo berriak biltzea da, atributu hori adierazten dutenak. Datu multzo hauek eskuragarri izan gabe kontrola nola lortu daitezkeen ikerkuntza-galdera irekia da, teknika berrien garapena eskatuko duena. Teknika hauek eta haiek garatzeko egindako ikerkuntzak sistema hauen funtzionamenduaren ulermena sakontzen eta haien inguruko ezagutza zabaltzen lagun dezakete.

Erabilgarritasun praktikoa. Interes zientifikotik at, garatutako teknikek erabilgarritasun praktiko argia izan dezakete. Izan ere, datu multzoak biltzea ikasketa sakoneko sistema baten garapen prozesuko zati neketsuena izan daiteke, gizaki adituen arreta eta denbora eskatzen baitu. Ondorioz, edozein atributu kontrolatu nahi denean datu multzo berriak biltzea ezinezkoa izan daiteke, edo eskuragarri dagoen datu multzoaren tamaina gehiegi urritu dezake. Ikasketa sakoneko ereduen kalitatea erabilitako datu multzoen tamainarekiko proportzionala izaten denez, komeni da hauen tamaina txikitzea saihestea. Bertso sortzailearen adibidera itzulita, askoz errazagoa izango da hizkuntza-ereduarentzat kalitatezko testua sortzen ikastea bertso mota guztietako testuak erabilita eredu bakar bat entrenatzen bada, atributuaren —bertso-egitura— aukera bakoitzeko eredu desberdin bat entrenatzen bada baino. Ondorioz, datu multzoak baldintzatu gabe sistemaren irteeraren atributuak kontrolatzeko teknikak garatzeak erabilera praktiko ugari izan ditzake.

1.2 Helburuak eta ikerketa-lerroak

Tesi honen helburua hizkuntzaren prozesamenduko sistemen irteeren atributuak kontrolatzeko teknika orokorrak garatzea da. Esparru orokor honen barruan, mota desberdinetako sistementzat garatu ditugu teknikak, eta atributu desberdinen kontrolagarritasuna aztertu dugu.

Tesiko lehen lanetan, nagusiki kontrolatu nahi den atributua adierazten duten berariazko datu multzoak behar ez dituzten metodo **ez-gainbegiratuetan** zentratu gara.[2] Tesiko azken zatian, berriz, eskala handiko sistema aurre-entrenatu baten irteeraren edozein atributu kontrolatu ahal izateko metodo orokor baten garapena ikertu dugu. Zehazki, tesiko ikerketa-lerroak hurrengo moduan egituratu dira:

[**L1**] **Atributuen kontrolagarritasunerako metodo ez-gainbegiratuen garapena**, hizkuntzaren prozesamenduko ataza desberdinetara aplikatua. Tesiaren garapenean zehar arloak bilakaera nabaria bizi izan du, eta bertan erabilitako teknikak eta arkitekturak aldatzen joan dira. Bilakaera honekin batera, atributuen kontrolagarritasuna ikertzeko erabili ditugun sistemak ere aldatzen joan dira. Tesiko lehen zati honetan hiru lan kokatu daitezke, eta bakoitzean atributu desberdin baten kontrolagarritasuna aztertzen da.

[**L1.1**] **Hitz-bektore estatikoen lerrokaketa geometrikoaren kontrolagarritasuna**, hitz-bektore elebidunen ikasketara aplikatua. Nahiz eta independenteki entrenatutako hizkuntza desberdinetako hitz-bektoreek egitura geometriko antzekoa izaten duten, haien lerrokaketa —hau da, orientazio geometrikoa— desberdina izaten da. Lan-lerro honetan lerrokaketa hau kontrolatzeko teknika bat garatu dugu, eta teknika hau artearen-egoerako hitz-bektore elebidunen ikasketa ez-gainbegiratura aplikatu dugu. Hitz-bektoreen inguruko aurrekarietarako, ikus 1.4.1 atala.

[**L1.2**] **Testuinguruaren araberako adierazpenen informazio edukiaren kontrola**, parafrasi-sorkuntzara aplikatua. Testuinguruaren araberako adierazpen-sistemek (§1.4.2), normalean adierazitako testuari buruzko informazio guztia biltzen dute. Hau da, posible da adierazpenean soilik oinarrituta jatorrizko testua berreskuratzea. Lan-lerro honetan, adierazpen hauek bildu dezaketen informazio edukia mugatzen dugu, adierazpen hauek testuaren azaleko propietateak —adibidez, sintaxiari buruzko xehetasunak— gorde ez ditzaten. Adierazpen hauek parafrasi-sorkuntzarako erabiltzen ditugu, eta arloan sarritan erabilitako erreferentziazko sistema bat gainditzen dugu.

[**L1.3**] **Hizkuntza-eredu batek sortutako testuaren metrikaren kontrola**, poesia-sorkuntzara aplikatua. Hizkuntza-eredu (§1.4.2) arruntek hitzez hitz sortzen dute testua, testuaren atributu globalen inongo kontrolik eskaini gabe. Lan-lerro honetan hizkuntza-eredu batek sortutako testuaren atributu globalak —guk aztertutako kasuan, metrika eta errima— kontrolatzeko metodo

[2] Adibidez, aurreko ataleko bertsoen adibidera itzuliz, zortziko txikiz osatutako testu-multzo batekin entrenatu gabe mota honetako bertsoak sortzeko gai den sistema bat ez-gainbegiratua dela esan genezake.

orokor bat garatu dugu, eta poesia-sortzaile bat entrenatzeko erabili dugu. Garatutako poesia-sortzailea egitura metriko eta errima zorrotzak jarraitzeko gai den lehen sistema neuronal ez-gainbegiratua da.

[L2] **Eskala handiko sistema aurre-entrenatuen kontrolagarritasuna**. Azken urteetan, hizkuntzaren prozesamenduko sistemen eskala izugarri handitu da —partikularki, hain arrakastatsuak izan diren hizkuntza-ereduen kasuan— bai ereduen tamainaren eta bai datu multzoen tamainaren aldetik. Eskala berri honek kalitate handiko sistemak sortzea ahalbidetu badu ere, hainbat arazo ekarri ditu (ikus 1.4.2 atala hauen eztabaida sakonago baterako). Adibidez, sistema hauek domeinu edo ataza berrietara egokitzeak konputazio kostu oso altua du, ikerlari eta erabiltzaile gehienek eskuragarri ez dutena. Gainera, sarritan sistema hauek kutxa-beltz moduan API baten bidez izaten dira eskuragarri, eta ezin da haien barne-funtzionamendua miatu edo aldatu. Gauzak honela, baliabide urrituko testuinguru batean sistema hauen portaera kontrolatu, edo domeinu eta ataza berrietara egokitu nahi izatekotan, teknika berriak garatu behar dira. Lan-lerro honetan eszenatoki honetan zentratzen gara, eta kutxa-beltz hizkuntza-ereduak egokitzeko teknika berri bat proposatzen dugu, ereduaren barne-funtzionamendua ezagutu gabe funtzionatu dezakeena.

1.3 Tesia osatzen duten artikuluak

Atal honetan tesia osatzen duten artikuluak aurkezten ditugu, eta bakoitza dagokion lan-lerroan eta tesiaren testuinguru orokorrean kokatzen dugu. Artikulu osoak A eranskinean aurki daitezke. Artikuluak irakurketa-orden gomendatuan aurkezten dira, aldi berean orden kronologikoarekin bat datorrena.

[A1] Ormazabal et al. (ACL 2021)

Aitor Ormazabal, Mikel Artetxe, Aitor Soroa, Gorka Labaka, and Eneko Agirre. 2021. Beyond offline mapping: Learning cross-lingual word embeddings through context anchoring. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6479–6489, Online. Association for Computational Linguistics.

Artikulu hau L1.1 lan-lerroan kokatzen dugu, eta bertan hitz-bektore elebidunak sortzeko metodo ez-gainbegiratu bat proposatzen da. Lan honen motibazioa gure aurreko Ormazabal et al. (2019) lanean egindako analisian oinarritzen da. Bertan, hitz-bektore elebidunak sortzeko bi metodo mota nagusiak alderatzen ditugu. Alde batetik, aldibereko metodoek bi hizkuntzetako hitz-bektoreak zuzenean espazio komun batean ikasten dituzte, baina horretarako corpus paraleloen gainbegiratze elebidun indartsua behar dute, askotan eskuragarri ez dagoena. Bestetik, mapaketa metodoek bi hizkuntzetako hitz-bektoreak independenteki ikasten dituzten, eta ondoren espazio komun batean lerrokatzen dituzte mapaketa pausu baten bidez. Lerrokaketa metodoek modu ez-gainbegiratuan funtzionatu dezakete, baina gure lanean ikusi genuen aldibereko metodoek kalitate hobea lortzen dutela.

Gauzak honela, gure helburua bi metodo klaseak ezkontzea da, datu paralelo elebidunak erabili gabe zuzenean espazio lerrokatu batean hitz-bektoreak ikasteko gai den metodo bat proposatuz. Lan honetan, frogatzen da hau posible dela, eta entrenamenduan aldaketa txiki batzuk eginez entrenatutako hitz-bektoreen lerrokaketa kontrolatzea posible dela. Lerrokaketa hau kontrolatuz, gure metodoak hitz-bektoreak zuzenean helburu-hizkuntza bateko hitz-bektoreen espazio berean ikastea ahalbidetzen du. Autoikasketa teknika bat baliatuz metodoak modu ez-gainbegiratuan funtzionatzea lortzen dugu, eta bi atazatan ebaluatuz frogatzen dugu lortutako hitz-bektoreak artearen-egoerakoak direla.

Artikulu hau ACL 2021 konferentzian argitaratu da, SCIE class 1 balorazioa duena.

[A1] Ormazabal et al. (ACL 2022)

Aitor Ormazabal, Mikel Artetxe, Aitor Soroa, Gorka Labaka, and Eneko Agirre. 2022b. Principled paraphrase generation with parallel corpora. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1621–1638, Dublin, Ireland. Association for Computational Linguistics.

Artikulu hau L1.2 lan-lerroan koka dezakegu, eta tesian eredu sortzaileak erabiltzen lehena izan da. Artikulu honetan, kodetzaile-deskodetzaile (§1.4.2) sistema baten tarteko adierazpenak gordetzen duen informazioa aztertu eta kontrolatzen dugu. Zehazki, itzulpen automatikorako entrenatutako sistema baten kasua hartzen dugu, eta tarteko adierazpenak jatorrizko esaldiari buruz gordetzen duen informazio edukia aztertzen dugu, informazio teoriaren ikuspegitik.

Informazio kopuru hori da hain zuzen ere lan honetan kontrolatu —zehazki, mugatu— nahi dugun atributua. Frogatzen dugu ikasketa prozesuan aldaketa txiki bat eginez —ikasketa-helburuan autokodetze[3] gaitasuna penalizatzen dugu— posible dela tarteko adierazpenak jatorrizko esaldiari buruz gordetzen duen informazio kopurua mugatzea, eta honetarako berme matematikoak diseinatu eta frogatzen ditugu.

Artikuluan, informazio edukiaren kontrolagarritasun hau parafrasi-sorkuntzara aplika-

[3] Autokodetzaile bat kodetzaile-deskodetzaile sistema bat da, non deskodetzailea tarteko adierazpenetik jatorrizko esaldia bera berreskuratzen saiatzen den

tzen dugu. Informazio mugatua duen tarteko adierazpenean oinarrituta deskodetzaile bat entrenatzen da jatorrizko esaldia berreskuratzen saiatzeko —hau da, autokodetzaile modura entrenatzen da— baina, informazio edukia mugatua denez, esaldi originala zehazki berreskuratu ordez, haren parafrasi bat sortzen da. Intuitiboki, itzulpen eredu baten adierazpenaren informazio edukia mugatuz, adierazpen honek itzultzeko beharrezkoa den informazioa —idealki, esaldiaren esanahia— mantentzen du, baina garrantzizkoa ez den azaleko formari buruzko informazioa "ahazten"du. Gainera, informazioaren gaineko kontrol honek parafrasi sistemaren portaera egokitzeko modu natural bat eskaintzen du: adierazpenean gero eta informazio gutxiago gorde, parafrasiek aniztasun handiago izango dute —hau da, jatorrizko esaldiarekiko desberdinagoak izango dira— eta gero eta informazio gehiago gorde, parafrasiak zehatzagoak izango dira. Ebaluazio automatiko eta giza-ebaluazioen bitartez frogatzen dugu gure sistemak literaturan sarritan erabilitako erreferentzia sistema batek baino parafrasi hobeak eta kontrolagarriagoak sortzen dituela.

Artikulu hau ACL 2022 konferentzian argitaratu da, SCIE class 1 balorazioa duena.

[A1] Ormazabal et al. (EMNLP Findings 2022)

Aitor Ormazabal, Mikel Artetxe, Manex Agirrezabal, Aitor Soroa, and Eneko Agirre. 2022a. PoeLM: A meter- and rhyme-controllable language model for unsupervised poetry generation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3655–3670, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Artikulu hau, L1.3 lan-lerroan kokatua, eredu sortzaileak —kasu honetan, hizkuntza-ereduak (§1.4.2)— erabiltzen bigarrena izan da tesian. Bertan *PoeLM* aurkezten da, poesia sortzeko gai den hizkuntza-eredu kontrolagarri bat. PoeLM poesia sortzeko gai da metrika eta errimaren kontrolagarritasunaren bidez: hizkuntza-eredu arrunt bati ez bezala, posible da gure ereduak sortzen duen testuak jarraitu behar dituen errima eta metrika patroiak zehaztea.

Kontrolagarritasun hau lortzeko, bi pausuko prozesu bat jarraitzen dugu. Lehenik, testu-corpus batetik interesatzen zaigun atributua erauzi eta etiketatzen da —kasu honetan, testu-corpus arrunt batean naturalki agertzen diren lerroen silaba kopuruak eta errimak— eta atributu hori kontrol-kodeetan biltzen da. Ondoren, kontrol-kode horiek testu soilarekin tartekatzen dira, testu aberastu bat sortzeko, eta hizkuntza-eredua testu aberastu honen gainean entrenatzen da. Hizkuntza-eredua entrenatu ondoren, prozesu hau alderantzikatu daiteke: sortu nahi dugun poesia motari dagokion errima eta metrika patroiak adierazten dituzten kontrol-kodeak ematen badizkiogu ereduari, ereduak kontrol-kodea errespetatu eta poesia mota hori sortuko du.

Baliatzen dugun kontrol-kodeen teknika, artikuluan poesia-sorkuntzarako soilik erabiltzen badugu ere, orokorragoa da; teorian, testutik automatikoki erauzi eta etiketatu daitekeen edozein atributu kontrolatzea ahalbidetu dezake. Izan ere, arloan kontrol-kodeen teknikak gure lanaren aurretik erabili izan dira (§1.5.3), baina gure artikulua egitura metrikoa bezalako atributu zurrun bat kontrolatzeko erabili daitezkeela frogatzen lehena izan da.

Kontrolagarritasun teknika honen bidez entrenatutako PoeLM eredua modu guztiz gainbegiratuan poesia egituratua sortzeko gai den lehen sistema neuronala da. Sortzen dituen poemen kalitatea neurtzen dugu, bai ebaluazio automatiko eta bai giza-ebaluazioaren bidez. Ebaluazio automatikoaren aldetik, frogatzen dugu sistema gai dela mota desberdinetako poemak sortzeko, soilik kontrol-kodeak aldatuz, haien errima eta metrika egiturak errespetatzen. Giza-ebaluazioan, gure sistemak gaztelaniaz sortutako poema motzak bolondresek 5 minututan sortutako poemekin alderatzen ditugu, eta ikusten dugu hainbat kasutan gizakiek gure sistemak sortutakoak nahiago dituztela.

Artikulu hau EMNLP 2022 konferentzian argitaratu da (Findings atalean), SCIE class 1 balorazioa duena.

[A1] Ormazabal et al. (EMNLP 2023)

Aitor Ormazabal, Mikel Artetxe, and Eneko Agirre. 2023. CombLM: Adapting black-box language models through small fine-tuned models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2961–2974, Singapore. Association for Computational Linguistics.

Tesiko azken artikuluan, L2 lan-lerroan kokatua, paradigma desberdin bat landu dugu. Aurreko lanetan atributu jakin bat kontrolatzeko metodo ez-gainbegiratuak —hau da, berariazko datuak behar ez dituztenak— proposatu baditugu, lan honetan hizkuntza-eredu handiak egokitzeko metodo orokor bat proposatzen dugu. Kasu honetan, suposatzen dugu ataza edo domeinu berri baterako datu multzoak baditugula, eta eredu aurre-entrenatu bat egokitu nahi dugula, ataza edo domeinu berrian eraginkortasuna hobetzeko.

Aurrekari (§1.4.2) eta erlazionatutako lanen (§1.5.4) ataletan azaltzen den bezala, hizkuntza-ereduak egokitzeko existitzen diren teknika gehienek konputazio kostu altuegia dute, bereziki egungo eredu handiekin lan egin nahi badugu. Gainera, kutxa-beltz ereduen kasuan, ezin da ereduaren barne-funtzionamendua miatu edo eraldatu, eta ondorioz ezin dira egokitze teknika klasikoak erabili.

Domeinu edo ataza berri baterako eredu bat lortu nahi bada, eredu handi bat egokitu ordez beste aukera bat eredu txiki bat gure datuekin zerotik entrenatzea, edo aurre-entrenatutako eredu txiki bat egokitzea da. Baina hurbilpen hau ere mugatua da, ezin baitu egungo eredu handien ezagutza orokor sakona baliatu.

Ondorioz, eredu aurre-entrenatu handiak domeinu eta ataza berrietara egokitzeko teknikak garatzea —ereduaren barne-funtzionamendua atzitu edo eraldatu gabe— gaur egun interes handiko ikerkuntza-lerro bat da. Artikulu honetan propietate hauek betetzen dituen egokitze teknika berri bat aurkezten dugu, *CombLM* izenekoa. Zuzenean eredu handia egokitu ordez, bi pausuko prozedura bat jarraitzen dugu: i) lehenik, eredu txiki aurre-entrenatu bat gure datuekin egokitzen dugu, eredu txiki "aditu"bat lortzeko, eta ii) eredu aditua eredu handi orokorrarekin konbinatzen dugu probabilitate-mailan, neurona-sare txiki baten bidez. Hurbilpen honek eredu handiaren ezagutza orokorra baliatu dezake, baina eredu txiki bat egokitzearen konputazio kostu baxua mantentzen. Gure metodoaren eraginkortasun eta orokortasuna frogatzen dugu, eredu handi bat domeinu eta ataza desberdinetara egokituz.

Artikulu hau EMNLP 2023 konferentzian argitaratu da, SCIE class 1 balorazioa duena.

1.4 Oinarriak

Atal honetan tesi honetako lana ulertu ahal izateko beharrezko aurrekariak aurkeztuko ditugu.

Tesiko helburu orokorra hizkuntzaren prozesamendu ereduen irteeraren atributu jakin batzuen kontrolagarritasuna lortzea izanda, lehenik kontrolatu nahi diren ereduak zehaztu behar dira. Tesiaren garapenean zehar, hizkuntzaren prozesamenduko arloaren bilakaera azkarra dela eta, artearen-egoerako ereduen arkitekturak aldatzen joan dira. Ondorioz, bilakaera honekin batera, garatutako kontrolagarritasun teknikak eredu klase desberdinetara aplikatu dira. Erabilitako ereduak bi kategoria orokorretan sailka daitezke: hitz-bektoreak eta eredu sortzaileak. Jarraian, sistema hauen deskribapena egingo dugu, haien funtzionamendua azalduz, eta bide batez arloaren bilakaeraren irudi orokor bat osatzeko eta eredu hauen agerpena bertan kokatzeko aprobetxatuz.

1.4.1 Hitz-bektoreak

Tradizionalki, ordenagailu batean, testu idatzia unitate diskretuen segida modura adierazi izan da: testua unitate atomikoetan banatzen da,[4] eta aurretik zehaztutako kodeketa baten bidez unitate hauek ordenagailuarentzat egokiak diren byte-formatura itzultzen dira, segida osoa adieraziko duen byte sekuentzia lortzeko. Sinbolo diskretuen segidatan oinarritutako adierazpen-sistema hauek, tradizionalki ordenagailuek izan dituzten beharretarako —nagusiki, testua gorde eta bistaratzea— eraginkorrak izan arren, hizkuntzaren prozesamendurako desiragarriak ez diren hainbat ezaugarri dituzte:

[4] Tradizionalki, unitate atomiko hauek hitzak izaten ziren, edo eskuz idatzitako erregelen bidez lortzen ziren. Hizkuntzaren prozesamenduko egungo ereduetan, ohikoa da hitzak automatikoki ikasitako azpi-hitz unitateetan banatzea, eta testua unitate atomikoetan banatzearen prozesu honi tokenizazioa deritzogu. Tokenizazioaren inguruko literatura-azterketa baterako, ikus Mielke et al. (2021).

- Ez dira *semantikoak*: Testu baten kodeketak ez du inongo loturarik testuaren esanahiarekin. Adibidez, unitate atomikoak karaktereak badira, *larri* eta *sarri* hitzen kodeketa askoz antzekoagoa izango da *maiz* eta *sarri* hitzena baino, nahiz eta azken biak sinonimoak diren. Beste modu batera esanda, adierazpenen antzekotasunak ez du esanahi semantikorik.

- Ez dira *egituratuak*: Adierazpenak, osatzen dituzten unitate atomikoak bezala, naturalki *diskretuak* dira, eta ezin da haien gainean inongo eragiketarik egin. Adibidez, ez du zentzurik bi hitzen adierazpenen baturaz edo batez bestekoaz hitz egiteak.

Ezaugarri hauek ikasketa sakoneko metodoekin ez dira ondo ezkontzen, ikasketa zaildu eta sistema hauen orokortze gaitasuna kaltetzen baitute. Arazo hauek ekiditeko, hizkuntzaren prozesamenduko arloan **hitz-bektoreak** erabili izan dira unitate atomikoak adierazteko.

Matematikoki, hitz-bektoreak unitate atomikoen eta $\mathbb{R}^N$ bektore-espazio euklidear baten arteko mapaketak dira: $\Phi : B \to \mathbb{R}^N$, non B unitate atomiko guztiez osatutako multzoa den. Hau da, mapaketa honek unitate bakoitzari N dimentsioko bektore bat esleitzen dio. Mapaketa hau adierazteko modu baliokide bat, atal honen gainontzekoan erabiliko duguna, $X \in \mathbb{R}^{|B| \times N}$ matrize bat definitzea da, unitate atomiko bezainbeste lerro izango dituena, non i. lerroak B-ko i. unitate atomikoari dagokion bektorea gordeko duen. Hitz-bektore hauek modu egokian entrenatzen badira, arestian aipatutako bi gabeziak ekiditen dituzte: espazio euklidearraren egitura dute, eta semantikoak dira, esanahi antzekoa duten hitzek elkarren arteko distantzia —normalean kosinu antzekotasunaren bidez neurtzen dena— txikia izaten baitute.

Hitz-bektoreen egitura —eta bilatzen ditugun propietateak— ezagututa, ikasketa metodoaren galdera geratzen da: nola entrenatu ditzakegu aipatutako propietateak betetzen dituzten kalitatezko hitz-bektoreak? Erantzuna **hipotesi distribuzional** delakoan oinarritzen da. Hipotesi honek zera dio: esanahi antzekoak dituzten hitzak testuinguru antzekoetan agertuko direla testu naturaletan, eta alderantziz (Harris; Firth, 1957). Hitz-bektoreak ikasteko teknika gehien-gehienak —eta atal honetan aurkeztuko diren guztiak— hipotesi honetaz baliatzen dira, testu naturalez osatutako corpus bateko hitzen agerkidetza patroietan oinarrituta hitzen adierazpenak ikasi ahal izateko.

Hitz-bektoreen ikasketa hizkuntzaren prozesamenduko arloan ibilbide luzeko ikerketa-lerroa da, eta metodo ugari existitzen dira. Metodo hauek bi multzotan sailkatu ohi dira (Miceli Barone, 2016a): kontaketetan oinarritutakoak, eta eredu prediktiboetan oinarritutakoak. Tesi honetan neurona-sareetan oinarritutako eredu prediktiboak erabili dira —azken hamarkadan gailendu direnak— eta hauetan zentratuko gara atal honetan. Kontaketetan oinarritutako ereduen ikuspegi orokor baterako, ikus (Turney and Pantel, 2010).

Eredu prediktiboen atzeko ideia orokorra hurrengoa da: neurona-sare bat entrenatzen badugu agerkidetza patroiak —hau da, hitz jakin baten inguruan beste zein hitz ager daitekeen— ikas ditzan, entrenatutako neurona-sarearen pisuetatik zuzenean hitz-bektoreak erauzi ahalgo ditugu. Hipotesi distribuzionalari esker, modu honetan entrenatutako eredu batetik erauzitako bektoreak naturalki semantikoak izango dira.

Eredu prediktiboak

Hitz-bektoreen ikasketaren inguruko literatura zabala da, eta ideia orokor hau jarraitzen duten teknika ugari existitzen dira. Izan ere, mende hasieratik neurona-sareen bidez hitzen adierazpenak ikastearen ideia jorratu zen (Bengio et al., 2000), eta adierazpen hauen ikasketa eta ataza desberdinetarako erabilgarritasuna hainbat lanetan aztertu zen (Collobert et al., 2011; Collobert and Weston, 2008; Turian et al., 2010; Huang et al., 2012). Baina hitz-bektoreen ikasketaren arloan entzute handia izaten lehen lana Mikolov et al. (2013c,a) izan zen, non hitz-bektoreak ikasteko **eredu log-linealetan** oinarritutako bi arkitektura berri proposatu ziren, modu azkar eta eraginkorrean hitz adierazpenak ikastea ahalbidetu zutenak. Eraginkortasun honek testu-corpus erraldoien gainean kalitatezko hitz-bektoreak entrenatzea posible egin zuen, **autogainbegiraketa printzipioa** jarraituz. Ondoren, hitz-bektore hauek ataza anitzetara aplikatu izan dira, kalitatezko sistemak lortzeko beharrezko berariazko datu kopurua asko txikituz.

Lan honetan proposatutako arkitektura hauek, zehazki *CBOW* eta *skip-gram*, izan dira tesian erabili izan direnak, eta atal honetan azalduko ditugu.

Eredu log-lineal hauek, testu-corpuseko hitz bat emanda, bere inguruan beste hitz jakin bat aurkitzearen probabilitatearen logaritmoa aurresaten ikasten dute. **Skip-gram** ereduak, hitz bat emanda, inguruko c hitzak aurresango ditu. Zehazki, honako ikasketa-helburua minimizatuko du:

$$\mathcal{L}_{SG} = -\sum_{t} \sum_{-c \leq j \leq c, j \neq 0} \log p(w_{t+j}|w_t),$$

non c testuinguru tamaina izango den, eta $\sum_t$ batukaria entrenamendu corpuseko hitz guztien gainean egingo den.

CBOW ereduak, berriz, prozesu hau alderantzikatu eta hitz baten testuingurua emanda, hitz hori aurresango du:

$$\mathcal{L}_{CBOW} = -\sum_{t} \log p(w_t|w_{t-c}, ..., w_{t-1}, w_{t+1}, ..., w_{t+c}).$$

Ikasketa-helburu hauek praktikan inplementatzeko, p probabilitateak parametrizatu behar dira. Horretarako, bi hitz-bektore multzo definitzen dira: $X \in \mathbb{R}^{|B| \times N}$ **sarrera bektoreak** —hitz-bektoreak definitzeko arestian deskribatutako matrize notazioa erabiliz—, eta $\tilde{X} \in \mathbb{R}^{|B| \times N}$ **irteera bektoreak**. Orduan skip-gram kasuan honela definitzen da probabilitatea:

$$p(w_{t+j}|w_t) = \frac{\exp(\hat{X}_{w_{t+j},*} \cdot X_{w_t,*})}{\sum_w \exp(\hat{X}_{w,*} \cdot X_{w_t,*})},$$

non X matrizeko i. lerroak, $X_{i,*}$, B-ko i. hitzaren hitz-bektorea gordetzen duen, $\cdot$ operadoreak biderketa eskalarra adierazten duen, eta izendatzaileko batura B-ko hitz guztien gainean egiten den.

Era berean, CBOW kasuan honela definituko litzake probabilitatea:

$$p(w_t|w_{t-c}, ..., w_{t-1}, w_{t+1}, ..., w_{t+c}) = \frac{\exp(\frac{1}{2c}\sum_{-c\leq j\leq c, j\neq 0} X_{w_{t+j},*} \cdot \hat{X}_{w_t,*})}{\sum_w \exp(\frac{1}{2c}\sum_{-c\leq j\leq c, j\neq 0} X_{w_{t+j},*} \cdot \hat{X}_{w,*})}.$$

Parametrizazio honek ematen die eredu hauei *log-lineal* izena, ekuazioen logaritmoa hartzean —exp terminoak desagertaraziz— aldagaien konbinaketa lineal bat geratzen baita.

Bai skip-gram zein CBOW ereduaren oinarrizko parametrizazioek izendatzailean B multzoko hitz guztien gaineko batura bat dute. Adierazpen mota hau —**softmax** izenekoa— oso erabilia da ikasketa sakoneko arloan probabilitate banaketa normalizatuak lortzeko, baina konputazionalki oso garestia izan daiteke, eta konputazio kostu honek testu-corpus handiekin entrenatzea eragotzi dezake. Arazo hau arintzeko, Mikolov et al. (2013c) lanean eraginkortasun konputazionala hobetzeko hainbat teknika aurkeztu ziren. Hemen tesian erabilitako bi ikusiko ditugu: (i) softmaxaren ordezkapen eraginkor bat, laginketa negatibo izenekoa, eta (ii) maiztasun handiko hitzen azpilaginketa, ikasketa prozesuaren eraginkortasuna hobetzeko teknika bat.

Laginketa negatiboan, ikasketa-helburuari aldaketa txiki bat egiten zaio. Hitz bat emanda, inguruko hitzak aurresaten ikasi ordez, emandako hitz bat ea testuingurukoa den edo ez aurresaten ikasten du. Skip-grami aplikatua, laginketa negatiboaren ikasketa-helburua honakoa da:

$$\mathcal{L}_{SGNS} = -\sum_t \sum_{-c\leq j\leq c, j\neq 0} \left(\log \sigma(\hat{X}_{w_{t+j},*} \cdot X_{w_t,*}) + \sum_{\substack{1\leq i\leq k \\ w_i \sim P_n}} \log \sigma(-\hat{X}_{w_i,*} \cdot X_{w_t,*}) \right),$$

non σ sigmoide funtzioa den, eta P_n zarata distribuzio bat den, ausazko hitz lagin negatiboak emango dituena. Sigmoidearen irteerak probabilitate bezala interpretatzen baditugu, adierazpen berriko bi zatiek interpretazio argia dute. Alde batetik, $\log \sigma(\hat{X}_{w_{t+j},*} \cdot X_{w_t,*})$ terminoak benetako testuinguruko hitzaren probabilitatea altua izatera bultzatzen du; bestetik, batukariko $\log \sigma(-\hat{X}_{w_i,*} \cdot X_{w_t,*})$ terminoek ausaz aukeratu diren lagin negatiboen probabilitatea txikia izatea sustatzen dute, intuitiboki ausaz aukeratutako hitz hauek testuinguruko hitza izateko probabilitate oso baxua izan beharko luketelako. Ausazko lagin negatibo hauek P_n zarata banaketa batetik lagintzen dira, aldez aurretik definitu beharko dena. Mikolov et al. (2013c) lanean honetarako unigrama banaketan oinarritutako probabilitate banaketa bat definitzen dute:

$$P_n(w) = \frac{f(w)^{3/4}}{\sum_{w'} f(w')^{3/4}},$$

non $f(w)$ w hitzaren maiztasuna den.

Maiztasun handiko hitzen azpilaginketak ikasketa prozesuari aldaketa txiki bat egiten dio, honen eraginkortasuna hobetzeko helburuarekin. Zehazki, entrenamendu corpuseko w_i hitz bakoitzari ezabatua izateko probabilitate bat esleitzen zaio, bere maiztasunaren araberakoa:

$$P(w_i) = 1 - \sqrt{\frac{t}{f(w_i)}},$$

non t aldez aurretik ezarritako atalase bat den, normalean 10^{-5} ingurukoa. Honela, proportzioan hitz usuenak gutxiago agertuko dira entrenamendu corpusean, eta hitz ezohikoak gehiago. Normalean maiztasun txikiko hitzek kalitatezko errepresentazioak ikasteko zailtasun handiagoak izaten dituztenez, aldaketa honek ikasketa prozesua azkartzen du.

Hitz-bektore elebidunak

Orain arte hitz-bektore elebakarrak —hizkuntza bakar bateko hitzak adierazten dituztenak— ikasteko teknikak aztertu ditugu. Hizkuntzaren prozesamenduko erabilera kasu askotan, baina, errepresentazio semantiko **elebidunak** —bi hizkuntzatako hitzak espazio komun batean adierazten dituztenak— izatea lagungarria da. Mota honetako adierazpen elebidunak erabili izan dira, adibidez, itzulpen automatiko ez-gainbegiraturako edo transferentzia-ikasketarako (Ruder et al., 2019b).

Nola lortu daitezke hitz-bektore elebidun hauek, orduan? Ideia "inozo"bat hizkuntza desberdinetako testu-corpus elebakarrak kateatu eta ikusitako eredu prediktiboak erabiltzea izango litzake. Baina, zoritxarrez, ideia honek muga argi bat du; eredu prediktiboak agerkidetza patroietan oinarritzen direnez, honela entrenatutako hitz-bektoreak ez ziren semantikoak izango: hizkuntza desberdinetako hitzek *testuinguru bereiziak* izango dituzte corpus kateatuan —ez dira elkarrekin agertuko— eta ondorioz ereduak ez luke hauen arteko harremana ikasiko.[5] Ondorioz, hitz-bektore elebidunak ikasteko eredu eta teknika propioak beharrezkoak dira.

Errepresentazio elebidunen ikasketa hizkuntzaren prozesamenduan luze jorratutako

[5] Praktikan, hizkuntza desberdinetako hitzak batzuetan testuingurua banatzen dute: corpus elebakarretan hizkuntza nahasketa neurri txiki batean aurkitzen da, eta hizkuntzen artean konpartitzen diren "aingura"hitz gutxi batzuk existitzen dira. Hala ere, faktore hauek soilik ez dira nahikoa hurbilpen arruntekin kalitatezko hitz-bektore elebidunak ikasteko.

gai bat da, kontaketetan oinarritutako hitz-bektoreen garaitik (Littman et al., 1998; Fung, 1997; Rapp, 1999). Hitz-bektore prediktiboen garaian, metodo hauek bi multzotan sailkatu ohi dira (Baroni et al., 2014): (i) aldibereko metodoak, eta (ii) lerrokaketan oinarritutako metodoak.

Aldibereko metodoek entrenamendu garaian bi hizkuntzetako hitz-bektoreak zuzenean bektore-espazio konpartitu batean ikastea dute helburu. Honetarako, arestian aipatutako testuinguru bereizien arazoa arintzeko, entrenamendu datu bereziak erabiltzen dituzte: corpus elebidun paraleloak (Luong et al., 2015; Gouws et al., 2015) edo hiztegi elebidunak (Duong et al., 2016) ohiko aukerak dira. Metodo hauek kalitate handiko hitz-bektoreak lortzen dituzten arren, gainbegiraketa indartsua —corpus elebidun paraleloak biltzea ez da erraza—behar izatearen desabantaila dute.

Lerrokaketa metodoek testuinguru bereizien arazoa modu guztiz desberdinean ekiditen dute. **Isometria hipotesi** (Miceli Barone, 2016b) delakoan oinarritzen dira: modu independentean entrenatutako hizkuntza desberdinetako hitz-bektoreak, nahiz eta "orientazio"berdina ez izan, egitura geometriko antzekoa izango dute, kontzeptu berdinak adierazten dituzten hitzen agerkidetza patroiak antzekoak baitira hizkuntza guztietan. Hipotesi honek ondorio argi bat dauka: posible izan beharko litzake transformazio lineal baten bitartez independenteki entrenatutako hizkuntza desberdinetako hitz-bektoreak espazio semantiko konpartitu batera mapatzea. Hau da, $X, Y \in \mathbb{R}^{|B| \times N}$ hizkuntza desberdinetako hitz-bektoreak badira, independenteki entrenatu direnak, posible izan beharko litzake $W \in \mathbb{R} \in N \times N$ matrize bat aurkitzea —mapaketa lineala adieraziko duena— non XW eta Y bektore-espazio semantiko berean lerrokatuak dauden. Lerrokaketa metodoen helburua, orduan, W transformazio hau ikastea da. Orokorrean, lerrokaketa metodoek gainbegiraketa elebidun askoz txikiagoa behar dute —izan ere, guztiz gainbegiratu gabeak izatera irits daitezke— baina aldibereko metodoak baino ahulagoak dira sortzen dituzten hitz-bektoreak.

Gure Ormazabal et al. (2019) lanean bi metodo klaseen arteko konparaketa zuzena egin genuen, ahulezia eta indar desberdinak dituztela ondorioztatuz. Tesi honetako lanetako batean hurbilpen hibrido bat proposatzen dugu, bien indarrak konbinatzeko helburuarekin.

1.4.2 Testuinguruaren araberako adierazpenak eta eredu sortzaileak

Aurreko atalean ikusitako hitz-bektoreek hizkuntzaren prozesamenduan paradigma aldaketa baten hasiera ekarri zuten, aurre-entrenatutako adierazpenen erabilera arloaren erdigunean kokatu baitzuten. Autogainbegiratze bidez entrenatutako hitz-bektoreek testu-corpus erraldoietatik entrenatutako informazio linguistiko guztia kodetu dezakete, eta ondoren informazio edo ezagutza hau beste hizkuntzaren prozesamendu sistema eta atazetan berrerabili daiteke. Ezagutza berrerabiltze hau ezinbestekoa da, hizkuntzaren prozesamendu ataza gehienentzat datu multzo handirik ez baita existitzen. Ideia orokor hau **transferentzia-ikasketaren** arloan kokatu daiteke, azken hamarkadan hizkuntzaren prozesamenduaren bilakaeran ezinbestekoa izan dena (Ruder et al., 2019a).

Aurre-entrenatutako adierazpenen erabilgarritasuna frogatzen lagundu zuten arren, ikusitako hitz-bektoreek gabezia nabariak dituzte. Nagusiki, ikasten dituzten adierazpenak **estatikoak** dira: esaldi batean *banku* hitza inguratzen duen testuingurua edonolakoa dela ere, hitz horri dagokion hitz-bektorea berdina izango da. Gizakiok egiten dugun hitzen erabilera, berriz, testuinguruaren araberakoa da: *banku* hitzak esanahi guztiz desberdina du *Bankura noa dirua ateratzera* edo *Bankuan eseriko naiz* esaldietan. Testuinguruarekiko mendekotasun honek **testuinguruaren araberako adierazpenen** beharra azpimarratzen du.

Gauzak honela, azken urteetan testuinguruaren araberako adierazpen aurre-entrenatuen ikasketarako eredu eta metodo ugari proposatu dira.

Hitz-bektoreen kasuan bezala, $w_1...w_n$ testu bat izanda —non w_i testua osatzen duen i. unitate atomikoa den— helburua w_i unitate bakoitzari $h_i \in \mathbb{R}^N$ adierazpen bektore bat esleitzea da. Hitz-bektoreen kasuan, mapaketa hau **estatikoa** da: Φ murgiltze funtzio batek definituko du w_i hitzaren bektorea, $h_i = \Phi(w_i)$, testuingurua kontuan hartu gabe. Testuinguruaren araberako adierazpenen kasuan, berriz, unitate bakoitzaren bektorea, unitate beraren eta inguruko unitateen araberakoa izango da, mapaketa **dinamiko** baten bidez:

$$h_1, ..., h_n = \Phi(w_1...w_n). \tag{1.1}$$

Mapaketa hau parametrizatzeko mota desberdinetako neurona-sareak erabili izan dira. Jarraian arloan garrantzi handikoak izan diren bi ikusiko ditugu: Neurona-sare errepikariak —ingelesez *recurrent neural network*, edo RNN— eta arreta-mekanismoan oinarritutako transformer arkitektura.

Neurona-sare errepikariak

Neurona-sare errepikariek modu **sekuentzialean** osatzen dituzte adierazpenak. Zehazki, i. posizioko adierazpena osatzean, w_i unitate atomikoa eta aurreko posizioko h_{i-1} adierazpena hartuko dira kontuan:

$$h_i = \mathrm{RNN}(h_{i-1}, w_i),$$

Honela, sare errepikakorrek testuinguruaren ezagutza baliatu dezakete adierazpena osatzean. RNN funtzioaren parametrizazioaren arabera, arkitektura desberdin asko definitu daitezke. Oinarrizko forma batean, $\mathrm{RNN}(h_{i-1}, w_i) = \sigma(WX_{w_i,*}, +Uh_{i-1} + b)$ parametrizazio sinple bat hartu daiteke, non X hitz-bektore estatiko matrize bat den. Praktikan, oinarrizko forma honen aldaketa ugari proposatu izan dira, adibidez, informazio jarioa kudeatzeko ateak gehitzen dituzten GRU eta LSTM sareak (Hochreiter and

Schmidhuber, 1997; Cho et al., 2014).

Oinarrizko forma honen beste ohiko aldaketa bat, geruza bakarra erabili ordez hainbat geruzatako sare errepikakor **sakonak** erabiltzea da. n geruzako sare batean, i. posizioko eta t. geruzako adierazpena, h_i^t, aurreko geruzako eta aurreko posizioko adierazpenetan oinarrituta osatuko da:

$$h_i^k = \mathrm{RNN}(h_{i-1}^t, h_i^{t-1}) = \sigma(Wh_i^{t-1} + Uh_{i-1}^t + b)$$

Lehen geruzako adierazpenak hitz-bektore estatikoen bidez adierazi ohi dira, $h_i^0 = X_{w_i,*}$, eta unitate bakoitzaren adierazpena azken geruzakoa izango da , $h_i = h_i^{n-1}$.

Testuinguruaren araberako errepresentazioen ikasketan entzute handiko lehen lanak sare errepikakorrak erabili zituzten, besteak beste Dai and Le (2015) laneko *semi-supervised sequence learning*, eta Peters et al. (2018) laneko ELMO. Hala ere, arkitektura errepikakorrak hainbat gabezia ditu: posizio bakoitzeko adierazpena bakarrik aurreko posizioko adierazpenean oinarrituta osatzen denez, ordura arte ikusitako sekuentzia osoko informazio guztia posizio horretako adierazpen bektorean gorde behar da. Honek testuan distantzia handiko dependentzia ikastea asko zailtzen du — teknikoki, **gradiente desagerkorren arazoa** izenekoa aurkitzen da— eta adierazpenak sekuentzialki osatu behar direnez, ikasketa prozesua ezin da modu errazean paraleloan exekutatu.

Arazo hauek ekiditeko hainbat arkitektura eta metodo proposatu izan dira, baina haien artean garrantzi handienekoa **arreta-mekanismoan** oinarritutako **transformer** arkitektura da.

Transformer arkitektura

Transformer arkitekturan (Vaswani et al., 2017), RNN-en mekanismo errepikaria arreta-mekanismoarekin ordezkatzen da. Arreta-mekanismoak kontsulta-bektore bat, $\mathbf{k} = k_1, ..., k_L$ gako-bektore sekuentzia bat, eta $\mathbf{v} = v_1, ..., v_L$ balio-bektore hartzen ditu, non $q, k_i v_i \in \mathbb{R}^N$, eta balio-bektoreen batez besteko haztatu bat itzultzen du:

$$attn(q, \mathbf{k}, \mathbf{v}) = \sum_{i=1}^{L} \alpha_i v_i$$

,

non α_i softmax funtzioaren bidez konputatzen den:

$$\alpha_i = \frac{\exp(\mathrm{score}(q, k_i))}{\sum_{j=1}^{L} \exp(\mathrm{score}(q, k_j))}.$$

score funtzioa bektoreen arteko antzekotasun neurri izan daiteke, ohizko bi aukera produktu eskalarra, $\mathrm{score}(q, k_i) = q \cdot k_i$, eta produktu eskalar egokitua —*scaled dot product* ingelesez– $\mathrm{score}(q, k_i) = \frac{q \cdot k_i}{\sqrt{N}}$ izanik, non N bektoreen dimentsioa den.

Intuitiboki, arreta-mekanismoak balio-bektore sekuentziako informazioa bektore bakar batean konbinatzen du, batez besteko haztatuaren bitartez. Gainera, bektore bakoitzari ematen zaion pisua —α_i elementuek zehazten dutena— modu malgu eta diferentziagarrian konputatzen da, gako- eta kontsulta-bektoreen arteko harremanaren arabera.

Transformer arkitektura arreta-mekanismo honetaz baliatzen da unitate atomikoen adierazpenak osatzeko. Zehazki, **auto-arreta** erabiltzen da, non gako, kontsulta, eta balio sekuentziak beti sekuentzia berdinetik datozen, aurretik transformazio lineal desberdinak aplikatu ondoren. Honela, n geruzako transformer batean, $w_1 ... w_n$ sarrera sekuentzia baldin badugu, RNN-en kasuan bezala lehen geruzako adierazpenak hitz-bektore estatiko batek definituko ditu, $h_i^0 = X_{w_i}, *$, eta ondoren geruza bakoitzeko adierazpenak aurreko geruzaren gainean auto-arreta eginez konputatuko dira:

$$h_i^t = \mathrm{LN}\left(h_i^{t-1} + f(attn(W_q^t h_i^{t-1}, W_k^t \mathbf{h}^{t-1}, W_v^t \mathbf{h}^{t-1}))\right),$$

non LN normalizazio funtzio bat den, f feedfoward neurona-sare bat den,[6] $W_q^t, W_k^t, W_v^t \in \mathbb{R}^{N \times N}$ proiekzio matrizeak diren, eta $W\mathbf{h}^k = Wh_1^t ... Wh_n^t$ matrize proiekzioa posizio guztietako adierazpenei aplikatzean lortzen den sekuentzia den. Aipagarria da ere gaur egun praktikan erabilitako transformer inplementazioek oinarrizko forma honi hainbat hobekuntza ezartzen dizkiotela, adibidez normalizazioa auto-arretaren aurretik egitea, edo arreta-mekanismoan hainbat proiekzio matrize erabiltzea.

Vaswani et al. (2017) lanak arreta-mekanismoan oinarritutako transformerra arloaren erdigunean kokatu zuen, eta gerora hizkuntzaren prozesamenduaren —eta adimen artifizialeko arlo gehienen— bilakaeran ezinbestekoa izan da. Izan ere, azken urteetan entzute handiko adimen artifizial sistema ia guztiek modu batean edo bestean arkitektura hau erabili dute. Enpirikoki emaitza oso onak lortzeaz gain, RNN-ekiko hainbat abantaila ditu transformer arkitekturak. Haien artean, bi nabarmendu ditzakegu: i) arreta-mekanismoa ikasketa prozesuan modu naturalean **paralelizatu** daiteke, milaka prozesagailuetan eredu erraldoiak entrenatzea ahalbidetzen duena, eta ii) arreta-mekanismoak posizio guztietako adierazpenak **zuzenean kontuan hartzen** ditu —RNNek ez bezala— eta honek ikasketa prozesua errazten du, arestian aipatutako gradiente desagerkorren arazoa arinduz.

Testuinguruaren araberako errepresentazioen ikasketan transformer arkitektura guztiz gailendu da azken urteetan. Arkitektura hau erabiltzen duten lehen lanen artean, GPT (Radford et al., 2018) eta BERT (Devlin et al., 2019) nabarmendu ditzakegu. Gainera, lan hauekin batera arloan garrantzi handiko beste paradigma aldaketa bat etorri zen. Errepresentazioen ikasketan lehen lanek —hitz-bektore estatikoak, edo ELMO testuinguruduna, adibidez— helburutzat ataza bakoitzerako entrenatuko diren sistemen **ezaugarri** bat izango ziren adierazpenak ikastea zuten. Adibidez, corpus handi

[6] Feedforward neurona-sare batek sarreraren transformazio linealak eta elementu-mailako funtzio ezlinealak tartekatzen ditu. Zehazki, Vaswani et al. (2017) transformer lan originalean, $f(h) = W_1 \max(0, W_2 h + b_2) + b_1$ forma erabiltzen dute, non max funtzioa elementuka aplikatzen den.

baten gainean entrenatutako hitz-bektoreak sentimendu-sailkapenerako entrenatutako neurona-sare baten sarrera adierazpenak izan daitezke. Lan berri hauetan, berriz, ohartu ziren posible zela ataza bakoitzerako entrenatutako berariazko sistemak **aurre-entrenatutako errepresentazio-sistemarekin guztiz ordezkatzea**, aukeran ataza bakoitzerako egokitze-pausu bat gehituz. Paradigma berri honetan, ataza bakoitzerako berariazko sistemak modu independentean entrenatu ordez, testu-corpus erraldoien gainean sistema bakar bat entrenatzen da, ezagutza orokorra bilduko duena, eta ondoren sare hau egokitu eta erabili daiteke ataza guztietarako. Honela, aurre-entrenatutako adierazpenak berariazko sistemen ezaugarri bat izatetik, sistema hauek guztiz ordezkatzera pasa ziren.

Ikasketa-helburuak

Testuingurudun araberako adierazpenen ikasketan erabilitako arkitekturak ikusi ditugu, baina sare neuronalak entrenatu ahal izateko, arkitekturaz gain **ikasketa-helburua** definitu behar dugu. Ikasketa-helburuak optimizazio prozesuan maximizatzen den funtzioa definitzen du, eta entrenamenduan zehar sarearen parametroak eraldatuko dira entrenamendu datuen gainean ikasketa-helburu hau ahalik eta altuena izateko. Arkitekturarekin eta entrenamendu datuekin batera, ikasketa sakoneko sistema baten portaera definituko du.

Testu-corpus erraldoietan oinarrituta adierazpenak modu eraginkorrean entrenatzeko ikasketa-helburu autogainbegiratu —hau da, testu hutsetik at inongo berariazko daturik behar ez dutenak— anitz proposatu izan dira. Haien artean gehien erabili izan diren biak azalduko ditugu: hizkuntza-eredu maskaratu ataza, eta hizkuntza-eredu ataza.

Hizkuntza-eredu maskaratu ataza kontzeptualki oso sinplea da: testu bat emanda, ausaz testu horretako hitz —edo unitate atomiko— batzuk ezabatzen dira, eta ereduak falta diren unitate horiek aurresaten ikasten du. Adibidez, suposatu transformer arkitektura erabiltzen ari garela, eta $\mathbf{w} = w_1 ... w_n$ sekuentzia dela gure entrenamendu testua. Ausaz, sekuentziako k posizio aukeratuko dira, $i_1 ... i_k$,[7] eta posizio horietako unitate atomikoak w_M maskara token berezi batekin ordezkatuko dira, $w_{i_j} \rightarrow w_M$, sekuentzia maskaratu berri bat lortzeko. Sekuentzia maskaratu hau transformer neurona-sarearen sarrera izango da, eta irteera sareak sortutako token adierazpenak izango dira, $h_1 ... h_n \in \mathbb{R}^N$. Token adierazpen hauetatik, sareak posizio bakoitzari dagokion tokena aurresaten saiatzen du. Horretarako, proiekzio lineal bat erabiltzen da, softmax normalizazio batekin jarraituz:

$$p_i(w_t) = \frac{\exp(W_{*,w_t} \cdot h_i)}{\sum_{l=1}^{|B|} \exp(W_{*,l} \cdot h_i)},$$

non $p_i(w_t)$ terminoak sekuentzia originaleko i. unitate atomikoa w_t izatearen probabilitatea adierazten duen, $W \in \mathbb{R}^{N \times |B|}$ proiekzio matrizea den, eta $|B|$ unitate atomikoen

[7] Ordezkatzen diren unitateen posizioa eta kopurua aukeratzeko laginketa estrategia anitz existitzen dira literaturan. Estrategia desberdinen azterketa baterako, ikus Yang et al. (2023).

multzoaren tamaina den. Orduan, ikasketa-helburua maskaratutako tokenak berreskuratzea da, hau da, ordezkatutako posizioak $i_1 ... i_k$ badira, $p_{i_j}(w_{i_j})$ probabilitateak maximizatu nahiko dira. Zehazki, log-probabilitateen batura, $\sum_{j=1}^{k} \log p_{i_j}(w_{i_j})$ maximizatuko da.

Intuitiboki, maskaratutako token baten h_i adierazpenetik posizio horretako unitate atomikoa aurresan ahal izateko, inguruko hitzen esanahia ulertu eta bildu beharko da, testuingurudun adierazpen batean. Ikasketa-helburu hau oso erabilia izan da testuingurudun adierazpenen ikasketako arloan, adibidez BERT ereduan (Devlin et al., 2019), eta emaitza enpiriko onak lortzen ditu.

Hala ere, azken urteetan erabiliena izan den ikasketa-helburua, **hurrengo hitz aurresate (HHA)** ataza, are eta sinpleagoa da. HHA atazan, neurona-sareak testu baten zati bat jaso, eta hurrengo hitza edo unitate atomikoa aurresaten ikasten du. Hau da, entrenamendu testua $\mathbf{w} = w_1 ... w_n$ baldin bada, k. unitate atomikoa aurreikustean, sarearen sarrera $w_1, ... w_{k-1}$ azpisekuentzia izango da, eta irteerako h_{k-1} adierazpenean oinarrituta aurresango da hurrengo unitate atomikoa, hizkuntza-eredu maskaratuen kasuan bezala proiekzio lineal bat eta softmaxa erabiliz:

$$p_k(w_t) = \frac{\exp(W_{*,w_t} \cdot h_k)}{\sum_{l=1}^{|B|} \exp(W_{*,l} \cdot h_i)},$$

non $p_k(w_t)$ k. unitate atomikoa w_t izatearen probabilitatea den. Orduan, ikasketa-helburua benetako tokenen log-probabilitateen batura, $\sum_{i=1}^{N} \log p_k(w_k)$, maximizatzea izango da. HHA ikasketa-helburuarekin entrenatutako ereduei **hizkuntza-ereduak** ere deritze.

HHA ikasketa-helburuarekin entrenatzean, predikzio pausu bakoitzean token bakar bat aurresaten da —hizkuntza-eredu maskaratuen kasuan ordezkatutako k token aurresaten ziren aldi berean— nahiz eta praktikan hainbat pausu paraleloan konputatzen diren neurona-sarearen exekuzio bakar batean, eraginkortasun konputazionala hobetzeko. Intuitiboki, h_k adierazpenean oinarrituta w_{k+1} hurrengo unitate atomikoa aurresan ahal izateko, sareak ikasi beharko du h_k adierazpenean $w_1 ... w_{k-1}$ azpisekuentzia osoko esanahia kodetzen. Gainera, sistemak modu honetan entrenatzeak albo-ondorio garrantzitsu bat du: ereduak testu baten zati bat ikusita hurrengo hitza aurresaten ikasten duenez, posible da eredu hauek erabiltzea **testu sorkuntza** egiteko. Sinpleki, ereduak aurresandako unitate atomikoa testu sekuentziari gehitu dakioke, eta ereduari berriz eskatu hurrengo unitatea aurresatea, iteratiboki testu oso bat sortu arte. Ikusiko dugun bezala, mota honetako **eredu sortzaileak** garrantzi handikoak izan dira arloaren bilakaeran.

HHA atazarekin entrenatutako lehen adierazpen aurre-entrenatuak (Radford et al., 2018) hizkuntza-eredu maskaratuen alternatiba bat ziren, baina paradigma orokor berdina jarraitzen zuten: lehenik adierazpen eredua testu kopuru handi batekin aurre-entrenatzen zen, eta ondoren ataza bakoitzeko egokitze prozesu bat jarraitzen zen, berariazko datuak erabiliz. Baina, aurre-entrenatzeko erabiltzen den testu-corpusen eta baliabide konputazionalen eskala handitu ahala, ikusi da posible dela **hizkuntza-ereduak ataza desberdinetara zuzenean aplikatzea**, zuzenean erantzuna sortzea eskatuz. Brown et al. (2020) lanean frogatu zuten hizkuntza-eredu handi hauek gai direla ataza berriak betetzeko, besterik gabe ataza horren adibideak sarrera sekuentzian bertan emanez. Gaitasun hau, *few-shot* ikasketa deritzona, HHA ataza ikastearen albo-ondorio modura hizkuntza-ereduek eskuratzen duten gaitasun berri adibide bat da. Gaitasun hauei **propietate emergenteak** deitu izan zaie (Wei et al., 2022). Paradigma berri honetan, ataza bakoitzerako egokitze prozesua ez da beharrezkoa, sinpleki ataza berri hori sarrera sekuentzian definitzearekin nahikoa delako. Azken urteetan, paradigma hau gailendu da arloan, eta artearen-egoerako eredu aurre-entrenatutako ia guztiak HHA ikasketa-helburuarekin entrenatutako hizkuntza-ereduak dira (Anil et al., 2023; OpenAI, 2023; Touvron et al., 2023).

Kodetzaile-deskodetzaile ereduak

Orain arte ikusitako arkitekturek sekuentzia bakar baten adierazpenak konputatzen dituzte, 1.1 ekuazioa jarraituz. Erabilera kasu batzuetan, ordea, bi sekuentzia ditugu, $x_1...x_n$ eta $y_1...y_n$, non bigarren sekuentziak lehenarekiko mendekotasuna duen, eta bien adierazpenak kalkulatu nahi dira, hurrengo ekuazioa jarraituz:

$$h_1^x, ..., h_n^x = \Phi^{enc}(x_1...x_n) \tag{1.2}$$

$$h_1^y, ..., h_m^y = \Phi^{dec}(y_1...y_m, h_1^x...{}_n^x) \tag{1.3}$$

Hau da, $x_1...x_n$ sekuentziaren adierazpenak sekuentzia bakarreko kasuan bezala kalkulatzen dira, baina $y_1...y_n$ sekuentziaren adierazpenak kalkulatzean, sarrera gehigarritzat lehen sekuentziaren adierazpenak jasotzen dira, $h_1^x...h_n^x$. Arkitekturako Φ^{enc} zatiari kodetzailea deritzo, eta Φ^{enc} zatiari deskodetzailea.

Kodetzaileak arestian deskribatutako sekuentzia bakarreko sistemen arkitektura bera erabili dezake, baina deskodetzaileak aldaketak behar ditu, sarrera gehigarri bat duelako. Transformerretan oinarritutako kodetzaile-deskodetzaile sistemetan, ohikoa da ikusitako auto-arreta mekanismoaz gain geruza bakoitzean **arreta-gurutzatuko** osagarri bat gehitzea, non kontsulta-bektoreak $y_1...y_m$ sekuentziaren adierazpenetik konputatzen diren, baina gako- eta balio-bektoreak kodetzaileak emandako $h_1^x...h_n^x$ adierazpenetatik kalkulatzen diren. Honela, bigarren sekuentziako adierazpenak osatzean transformerrak lehen sekuentziako informazioa ere barneratu dezake.

Mota honetako arkitekturei **kodetzaile-deskodetzaile** deritze, eta oso erabiliak dira *sequence-to-sequence* motako atazetan, non bai sarrera eta bai irteera sekuentziak diren. Besteak beste, itzulpen automatikoan, parafrasi-sorkuntzan eta laburpen sistemetan erabili izan dira. Tesi honetan parafrasi-sorkuntzarako kodetzaile-deskodetzaile sistema bat erabili dugu, 1.2 ataleko **L1.2** lan-lerroan.

Hizkuntza-ereduak, eskala, eta kutxa-beltz ereduak

Arestian deskribatutako eredu aurre-entrenatuen —partikularki, hizkuntza-eredu sortzaileen— arrakastaren atzeko eragile nagusia **eskala** izan da: HHA atazarekin eredu gero eta handiagoak [8] gero eta testu-corpus handiagoekin entrenatzean, eredu hauek gaitasun berriak eskuratzen dituzte. Honen ondorioz, gaur egun eskuragarri ditugun eredu onenak **izugarri handiak** dira —milaka milioi parametro dituzte—, eta hauek exekutatzeak **konputazio kostu altua** dakar. Gainera, industria aktore askok ez dituzte haiek sortutako ereduen parametroak publikoki eskuragarri jarri, soilik ereduak API[9] baten bidez eskuragarri jartzen. Testu-sarrera bat jaso eta irteera bat ematen duten —barneko neurona-sarearen xehetasunak funtzionamendua miatu edo eraldatu ahal izan gabe— eredu hauei **kutxa-beltz** ereduak deritze.

Nahiz eta egungo eredu aurre-entrenatuak gai diren hainbat ataza zuzenean betetzeko, oraindik sarritan komenigarria da eredu hauek ataza berrietara zuzenean **egokitzea**. Hizkuntza-eredu handien agerpenaren aurretik erabilitako egokitze teknika gehienek eredua entrenatzen jarraitzen dute (Ruder et al., 2019a), eta ondorioz ez da posible teknika hauek erabiltzea kutxa-beltz ereduak —edo konputazio kostuagatik entrenatzeko handiegiak diren ereduak— egokitzeko. Ondorioz, azken urteetan **kutxa-beltz ereduen egokitzearen** inguruko hainbat lan argitaratu dira, barne-funtzionamendua ezagutu gabe eredu bat egokitzeko teknikak proposatzen dituztenak.

1.5 Erlazionatutako lana

Atal honetan tesiko lan-lerro desberdinekin erlazionatuta dagoen literatura aurkeztuko dugu. Tesiaren helburu orokorra —atributuen kontrolagarritasuna— artikulu guztietan agertu den arren, lan bakoitzean ideia orokor honek ahalbidetu duen aplikazioa aldatzen joan da, hitz-bektoreen ikasketatik hizkuntza-ereduen egokitzera. Ondorioz, atal hau aplikazioen arabera antolatuko dugu, lan-lerro desberdinek jorratzen dituzten atazen inguruko literatura aurkezten. Zehazki, lau ataletan banatuko dugu literatura-azterketa: hitz-bektore elebidunen ikasketa (§1.5.1), parafrasi-sorkuntza (§1.5.2), poesia-sorkuntza (§1.5.3) eta kutxa-beltz hizkuntza-ereduen egokitzea (§1.5.4).

[8] Tesi hau hasi zen urtearen hasieran, hizkuntzaren prozesamenduan argitaratutako eredu handiena 11 mila milioi parametroko T5 eredua zen (Raffel et al., 2023). Gaur egun 170 mila milioi parametro baino gehiagoko eredu ireki ugari existitzen dira (Workshop et al., 2023; Almazrouei et al., 2023).

[9] API —ingelesez *application programming interface*— bat programa batek kanpoko softwarea atzitzea ahalbidetzen duen interfaze bat da, ikus `https://en.wikipedia.org/wiki/API`.

1.5.1 Hitz-bektore elebidunen ikasketa

Aurrekarietan azaldu den bezala (§1.4.1), hitz-bektoreen ikasketarako metodo gehienek hizkuntza bakar bateko adierazpenak ikas ditzakete, baina aldaketak egin gabe ezin dira testuinguru elebidunean erabili. Hala ere, hitz-bektore elebidunen ikasketa ibilbide luzeko ikerkuntza-lerroa da, ia hitz-bektore elebakarrekin batera aztertzen hasi zena. Kontaketan oinarritutako metodoen garaitik ikasketa elebidunerako teknikak proposatu izan badira ere (Littman et al., 1998; Fung, 1997; Rapp, 1999; Garera et al., 2009; Gaussier et al., 2004; Peirsman and Padó, 2008), tesi honetan erabili diren —eta azken urteetan gailendu diren— **eredu prediktibo** bidezko hitz-bektoreentzat proposatu diren metodoetan zentratuko gara. Hauek bi multzo nagusitan sailka daitezke: ikasketa prozesua bera hedatzen dutenak, eta lerrokaketa metodoak.

Ikasketa prozesua hedatzen duten metodoak izan ziren hitz-bektore prediktiboen hastapenean arreta handiena jaso zutenak. Metodo hauen helburua zuzenean hizkuntza desberdinetako hitz-bektoreak espazio banatu batean ikastea da, eta horretarako normalean ikasketa-helburuan aldaketa bat egiten dute, datu multzo paraleloen laguntzaz eleaniztasuna —hau da, bi hizkuntzetako hitz-bektoreak espazio semantiko berean lerrokatuak egotea— bermatzen duena. Hurbilpen arrakastatsu bat skip-gram (§1.4.1) ikasketa-algoritmoa hedatzea da, lerrokatutako dokumentu paraleloetaz baliatzeko; ikasketa-helburu hedatuak, hitz bakoitzaren adierazpena bere testuinguruko hitzen adierazpenetik gertu egoteaz gain, beste hizkuntzan dagokion hitzaren testuinguruko hitzetatik ere gertu egotea sustatzen du (Gouws et al., 2015; Luong et al., 2015; Coulmance et al., 2015). Skip-gram ordez beste arkitekturatan oinarritutako metodoak ere proposatu izan dira (Klementiev et al., 2012; Kočiský et al., 2014; Lauly et al., 2014). Dokumentu paraleloak ordez beste motako gainbegiraketa elebiduna erabiltzen dituzten metodoak ere proposatu izan dira, besteak beste hiztegi elebidunak (Gouws and Søgaard, 2015; Duong et al., 2016), eta dokumentu konparagarriak (Vulić and Moens, 2015; Vulic and Moens, 2016). Hala ere, metodo hauen ezaugarri bereizgarrietako bat gainbegiraketa elebidun garestiaren beharra da, eta hain zuzen ere hau da lerrokaketa metodoekiko duten desabantaila nagusia. Multzo honetako lanen inguruko ikuspegi zabalago baterako, ikus Ruder et al. (2019c).

Aldibereko metodoei kontrajarrita **lerrokaketa metodoen** multzoa dugu. Metodo hauek hizkuntza bakoitzeko —jatorrizko hizkuntza bat, eta helburu-hizkuntza bat— hitz-bektoreak modu arruntean ikasten dituzte, eredu elebakarrak erabilita, eta ondoren **mapaketa** bat ikasten dute —jatorrizko hitz-bektoreei, helburuko hitz-bektoreei, edo biei aplikatu dakiokeena— hitz-bektoreak espazio komun batera eramaten dituena. Paradigma orokor honek lerrokaketa metodo guztiak biltzen dituen arren, mapaketa ikastean datza konplexutasuna, eta hau egiteko modu desberdinak proposatzen dituzten lan pila argitaratu izan dira. Hauetatik lehena Mikolov et al. (2013b) izan zen, hitz baten mapatutako bektorearen eta haren itzulpenaren —hiztegi elebidun batek ematen ditu hitz-itzulpen pareak— bektorearen arteko L2 distantziaren karratua minimizatuz mapaketa lineal bat ikastea proposatu zuena. Lan honek frogatu zuen posible zela hiztegi elebidun batez baliatuta modu independentean entrenatutako hitz-bektoreak lerrokatzen zituen mapaketa lineal bat ikastea, eta proposatutako ikasketa-helburua ondoren etorri ziren lan ugariren oinarri izan zen. Lerrokaketaren kalitatea hobetzeko helburuarekin, oinarrizko ideia honen hedapen eta aldaketa ugari proposatu ziren hurrengo urteetan: L2 normalizazioa gehitzea ikasketa-helburuan (Dinu et al., 2015), korrelazio kanonikoaren analisia erabiltzea mapaketa ikasteko (Faruqui and Dyer, 2014), edo mapaketa linealari ortogonaltasun murriztapen bat ezartzea (Xing et al., 2015; Artetxe et al., 2016), besteak beste. Hasiera batean hedapen guzti hauek haien artean independenteak ziruditen arren, Artetxe et al. (2018a) lanean denak orokortzen dituen metodo orokor bat proposatu zen, arestian aipatutako teknika guztiak metodo honen kasu bereziak zirela frogatuz.

Lerrokaketaren kalitatea hobetzetik at, **gainbegiratze maila** murriztea helburu duen lan-lerroak ere arreta handia jaso du. Izan ere, orain arte aipatutako lan guztiek tamaina handiko hiztegi elebidunak erabiltzen dituzte —gutxienez 5000 hitzekoak— hizkuntza pare askorentzat eskuragarri ez daudenak, eta hiztegi txikiegiak erabiltzean emaitza kaskarrak lortzen dira (Vulić and Korhonen, 2016). Dokumentu mailan-lerrokatutako corpus elebidunak eskuragarri izatearen kasuan, Vulić and Korhonen (2016) frogatu zuten posible zela bi pausuko prozesu bat jarraitzea, lehenik aldibereko metodo baten bidez hitz-bektoreak entrenatuz, eta ondoren hitz-bektore horien bidez erauzitako hiztegi elebidun bat erabiliz kalitatezko mapaketa bat ikasteko.

Baina agian ikerkuntza-lerro honetan arrakasta handieneko lanak **autoikasketan** oinarritutakoak izan dira. Metodo hauek hazi-hiztegi txiki edo zaratatsu bat erabiltzen dute lehen mapaketa bat ikasteko, eta ondoren lerrokatutako hitz-bektoreak erabiltzen dituzte hiztegi hobe bat erauzteko. Prozesu hau iteratiboki errepikatzen da, hiztegiaren hobetzea eta mapaketa berri baten ikasketa txandakatzen. Artetxe et al. (2017) lanean proposatu zen lehen hurbilpen bat, frogatuz posible zela kalitatezko mapaketak ikastea soilik 25 hitz-pareko hazi-hiztegi elebidun bat erabiliz, eta Hauer et al. (2017) lanean ere antzeko *bootstrapping* metodo bat proposatu zen.

Gainbegiratze maila murriztetik at, ez askoz aurrerago gainbegiratze elebiduna **guztiz ezabatzea** posible zela ere frogatu zen. Aurretik marko ez-gainbegiratu honetan hitz-bektoreen lerrokaketa aztertu zuten lanak egon baziren ere (Miceli Barone, 2016b; Zhang et al., 2017b), baldintza estandarretan arrakasta izaten lehenak Lample et al. (2018) lana izan zen. Bertan, autoreek lehenik ikasketa antagonikoa erabiltzen dute hazi-hiztegi bat lortzeko, eta ondoren arestian aipatutako autoikasketa estrategia uztartzen dute kalitatezko mapaketa batera iristeko. Artetxe et al. (2018a) lanean ere euren aurreko laneko autoikasketa metodoa uztartzen dute —hobekuntza batzuekin— baina ikasketa antagonikoa ordez hizkuntza bakoitzeko hitzen antzekotasun-banaketetan oinarritutako **heuristikoez** baliatzen dira hazi-hiztegia lortzeko. Metodo ez-gainbegiratuen lehen arrakasta hauen ostean ikerkuntza-lerro honek arreta handia jaso zuen, eta metodo berri pila proposatu ziren (Yang et al., 2018; Hoshen and Wolf, 2018; Zhang et al., 2017c; Grave et al., 2018).

Lerrokaketa metoden abantailen ondorioz —hizkuntza bakoitzeko hitz-bektore elebakar aurre-entrenatuekin lan egiteko aukera, eta gainbegiratze maila baxua, nagusiki— aldibereko metodoen aldean gailendu badira ere, badituzte **muga** garrantzitsuak. Arestian aipatutako metodo guztiek mapaketa linealak erabiltzen dituzte, eta erabaki honek suposizio inplizitu ba egiten du: independenteki entrenatutako hizkuntza bakoitzeko hitz-bektoreen egitura geometrikoa antzekoa dela, eta transformazio lineal bat nahikoa dela biak lerrokatzeko. Suposizio honi **isometria edo isomorfismo** hipotesia deitu izan zaio, eta haren baliozkotasuna eta ondorioak aztertzen dituzten hainbat lan argitaratu izan dira. Nakashole and Flauger (2018) lanean hitz-bektore espazioko azpieremu desberdinek mapaketa lineal desberdinak behar dituztela erakutsi zuten, hitz-bektoreen arteko erlazio globala ez-lineala dela —eta ondorioz, espazioak benetan isomorfikoak ez direla— ondorioztatuz, eta euren ondorengo Nakashole (2018) lanean azpieremuetan oinarritutako lerrokatze metodo berri bat proposatu zuten. Søgaard et al. (2018) lanean hitz-bektore desberdinen isometria maila **zuzenean neurtzea** proposatu zuten. Horretarako metrika automatiko bat proposatu zuten, eta frogatu zuten metrika horrek korrelazio estua zuela Lample et al. (2018) laneko metodo ez-gainbegiratuak lortutako emaitzen kalitatearekin. Patra et al. (2019) lanean Gromov-Hausdorff distantzian oinarritutako beste isometria metrika bat proposatu zen. Metrika hauen eta mapaketen kalitatearen arteko korrelazio honek metodo hauek egiten duten suposizio inplizitua betetzearen garrantzia azpimarratzen du: isometria maila baxua denean, metodo hauek ez dute ondo funtzionatzen. Isometria hipotesia bereziki garrantzitsua da metodo ez-gainbegiratuen kasuan, gainbegiratze elebiduna izan gabe soilik egitura geometrikoan oinarritzen baitira mapaketa ikasteko. Lan-ildo berean, Vulić et al. (2019) lanean hizkuntza pare desberdinetarako Artetxe et al. (2018b) metodoa —euren aurreko lan baten arabera egonkorrena dena— aplikatu zuten, eta erakutsi zuten sarritan metodo ez-gainbegiratuak huts egiten duela, partikularki hizkuntza pareak linguistikoki urrunak direnean.

Muga hauen aurrean, naturalki galdera bat etortzen da: isometria-ezaren ondorioz agertzen diren arazo hauek hizkuntza desberdinetako hitz-bektoreak espazio komun batean lerrokatzen saiatzearen ondorio saihestezin bat al dira, edo soilik lerrokaketa metodoen muga bat? Gure tesi aurreko Ormazabal et al. (2019) lanean galdera hau ezezkoan erantzuten dugu, frogatuz **aldibereko metodoek hitz-bektore elebidun isometrikoagoak eta kalitate hobekoak** sortzen dituztela. Gauzak honela, bi metodo klaseen abantailak —aldibereko metodoen kalitatea eta lerrokatze metodoen gainbegiratze behar txikia— ezkontzea helburu zuen ikerkuntza-lerro bat ireki zen. Cao et al. (2016) lanean aldibereko metodo ez-gainbegiratu bat proposatu zen, baina baldintza berezietan baino ez zuten ebaluatu, eta ez zuen jarraipenik izan. Wang et al. (2020) lanean, berriz, metodo hibrido bat proposatu zen, lehenik corpus elebakarren kateaketaren gainean algoritmo elebakar baten bidez hitz-bektoreak ikasten dituena, eta ondoren lerrokatze mapaketa bat ikasten duena. Lan-lerro hau jarraitu genuen gure Ormazabal et al. (2021) lanean, aldibereko metodoen eta lerrokatze metodoen arteko erdi-puntu bat proposatuz. Gure metodoan, lehenik helburu-hizkuntzako hitz-bektoreak metodo elebakar baten bidez ikasten dira, eta ondoren jatorri-hizkuntzako hitz-bektoreak entrenatzen dira skip-gram algoritmoaren hedapen baten bidez, non testuinguruko hitz batzuen adierazpenak haien ordainen helburu-hizkuntzako adierazpenekin ordezkatzen diren. Ordezkatutako helburu-hizkuntzako bektore hauek aingura modura jokatzen dute, entrenatutako hitz-bektoreak helburu-hizkuntzarekin lerrokatuta egotea sustatzen. Ordainak lortzeko hiztegi elebidun bat behar da, baina heuristikoei eta lerrokatze literaturan inspiratutako autoikasketa teknika bati esker hau **modu ez-gainbegiratuan** lortzen dugu.

Gerora argitaratutako lanetan isometria arazoa arintzen duten beste hainbat metodo proposatu izan dira. Marchisio et al. (2022) lanean skip-gram algoritmoa hedatzen dute helburu-espazio batekiko isometria zuzenean optimizatzeko, mapaketa linealen kalitatea hobetzeko helburuarekin. Ganesan et al. (2021) metodoan, isometrikoagoak diren hitz-bektore elebakarrak sortzen saiatu ordez, mapaketa ez-lineala ikasten duen metodo bat proposatzen dute, isometrikoak ez diren hitz-bektoreak lerrokatu ahal izateko motibazioarekin.

1.5.2 Parafrasi-sorkuntza

Parafrasi-sorkuntzak ibilbide luzea du hizkuntzaren prozesamenduko arloan, metodo estatistikoen garaitik. Lehen ahaleginak nagusiki **erauzketan** oinarritzen ziren, hau da, esaldi bat emanda zuzenean haren parafrasi bat sortu ordez, aldez aurretik zehaztutako corpus batean parafrasiak *identifikatzen* dituzten sistemak ziren. Barzilay and McKeown (2001) lanean dokumentu paraleloen corpus elebakar bat erabili zuten parafrasiak identifikatzeko sailkatzaile ez-gainbegiratu bat entrenatzeko, eta corpus paraleloak baliatzen zituzten beste hainbat lan argitaratu ziren garai hartan (Pang et al., 2003; Shinyama et al., 2002; Dolan et al., 2004). Corpus paralelo elebakarrez gain beste motako baliabideak ere erabili izan ziren. Barzilay and Lee (2003) lanean corpus ez-paraleloak baliatu zitzakeen metodo bat proposatu zuten, eta beste lan batzuetan corpus elebakarrak ordez corpus paralelo elebidunak —hau da, itzulpen corpusak— erabili ziren, lortzeko askoz errazagoak direnak (Bannard and Callison-Burch, 2005). Garai honetako lan gehienak parafrasien identifikazioan zentratutako erauzketan oinarritutako sistemak izan arren, lan batzuetan parafrasi **sorkuntza** ere aztertu zen (Barzilay and Lee, 2003; Bannard and Callison-Burch, 2005).

Ikerkuntza-lerro honek bere jatorria erregeletan eta metodo estatistikoetan badu ere, gaur egun artearen-egoerako parafrasi sistema guztiak neurona-sareetan oinarritzen dira. Gainera, erabilitako **gainbegiratze motaren** arabera sistemen arkitekturak eta entrenamendu teknikak oso desberdinak izan daitezke. Hiru multzo orokorretan sailka ditzakegu lan gehienak, gainbegiratze maila txikienetik indartsuenera: (i) corpus elebakarrak erabiltzen dituztenak, (ii) corpus paralelo elebidunak erabiltzen dituztenak, eta (iii) parafrasi corpusak erabiltzen dituztenak.

Corpus elebakarrak soilik erabiltzen dituzten metodoak dira gainbegiratze behar txikiena dutenak. Tamaina eta kalitate handiko corpus elebakarrak lortzea erraza da gaur egun baliabide handi eta ertaineko hizkuntzentzat, beste baliabide motak —parafrasi corpusak, adibidez— ez bezala, eta ondorioz mota honetako metodoak bereziki erakargarriak izan daitezke. Adibide esplizituririk gabe parafrasiak sortzen ikasteko, proposatutako sistema gehienek kodetze-deskodetze arkitektura bat jarraitzen dute, eta **informazio-mugatzearen** teknika baliatzen dute. Ideia orokorra kodetzaileak esaldia adierazpen batean kodetzea, eta deskodetzaileak adierazpen horretatik jatorrizko esaldia berreskuratzea da, autokodetze egitura jarraitzen. Besterik gabe entrenatuta, honelako sistema batek jatorrizko esaldia kopiatzen ikasiko luke, baina adierazpen horren informazio edukia mugatzen bada, **jatorrizko esaldiari buruzko informazioaren zati bat "ahaztuko"da**, eta deskodetzaileak sortutako esaldia ez da jatorrizkoaren berdin-berdina izango. Bowman et al. (2016) lanean autokodetzaile bariazional bat erabiltzen dute parafrasi identifikaziorako, eta Roy and Grangier (2019) autokodetzaile bariazional kuantizatu bat erabiltzen da parafrasi-sorkuntzarako, non kuantizazioak adierazpenaren informazio edukia mugatzen duen. Huang and Chang (2021) lanean informazio mugatuko hitz-zaku adierazpen bat erabiltzen dute, eta parafrasia sortzean egitura sintaktikoa esplizituki modelatzen dute. Li et al. (2018) lanean informazio-mugatzearen paradigma alde batera uzten dute, eta errefortzu-ikasketa erabiltzen dute deskodetzaileari jatorri esaldia ez kopiatzen irakasteko.

Gainbegiratze mailaren beste muturrean, **parafrasi corpusak** erabiltzen dituzten sistemak ditugu. Mota honetako corpus handiak eskuratzea zaila izaten da, baina ikasketa gainbegiratua erabiliz sistemek **parafrasi ataza zuzenean ikastea** ahalbidetzen dute. Hau da, posible da kodetzaile-deskodetzaile sistema bat entrenatzea zuzenean parafrasi corpusaren gainean, parafrasiak sortzen ikas ditzan. Hurbilpen hau jarraitzen duten hainbat lan argitaratu dira, kodetzaile-deskodetzailerako arkitektura desberdinak erabiltzen, bai LSTM errepikariak (Prakash et al., 2016; Gupta et al., 2018; Kumar et al., 2019), eta aurrerago transformer arkitektura (Egonmwan and Chali, 2019).

Beste lan-lerro batek parafrasi corpusak erabiltzen ditu autokodetzaile motako sistemak modu berezian entrenatzeko, non adierazpenean *esanahia* eta *azaleko forma* esplizituki banatzen diren, ondoren esanahi bereko baina azaleko forma desberdineko parafrasiak sortu ahal izateko (Chen et al., 2019; Kumar et al., 2020; Hosking and Lapata, 2021). Hauen artean lehen bi lanek parafrasia sortzean forma sintaktikoa aldez aurretik zehaztea eskatzen dute, baina Hosking and Lapata (2021) lanean behar hau ekiditen dute, azaleko forma automatikoki aurresateko modulu bat entrenatuz.

Arestian aipatu dugun bezala, parafrasi sistema mota honen muga nagusia **parafrasi corpusen tamaina eta kalitatean** datza. Lan hauetan erabilitako parafrasi corpus publikoak, besteak beste *MSCOCO* (Lin et al., 2014), *WikiAnswers*, *Twitter* (Lan et al.,

2017) eta *Quora Question Pairs*[10] ingelesezkoak dira, haien tamaina mugatua da, eta parafrasien kalitatea sarritan txarra izaten da. Gauzak honela, orain arte ikusitako bi sistema klaseen arteko erdi-puntu bat erabilgarria izango litzateke.

Erdi-puntu honetan aurkitzen dira hain zuzen ere **corpus paralelo elebidunak** —hau da, bi edo hizkuntza gehiagotako corpus lerrokatuak, bata bestearen itzulpenak direnak— baliatzen dituzten metodoak. Mota honetako corpusak parafrasi corpusak baino askoz ugariagoak dira, eta kalitate handiagokoak, baina aldi berean corpus elebakar arruntek baino gainbegiratze seinale indartsuagoa eskaintzen dute. Klase honetako metodo gehienek **itzulpen automatikoa** baliatzen dute, Bannard and Callison-Burch (2005) lanaren garaitik proposatu izan den hurrengo intuizioa jarraituz: hizkuntza bateko bi esaldi, e_1 eta e_2, beste hizkuntza bateko f esaldi berdinera itzultzen badira, orduan bi esaldiek esanahi bera izango dute. Posible da, orduan, e_1 esaldia f *pibote* batera itzultzea, eta pibote hori bueltan itzultzea jatorrizko hizkuntzara, parafrasi bat lortzeko. Mallinson et al. (2017) ideia hau hedatzen dute, pibote bakar bat ordez hainbat itzulpen-pibote erabiliz, eta Hu et al. (2019) lanean murriztapen lexikoak gehitzen dituzte itzulpen prozesuan, parafrasien kalitatea hobetzeko helburuarekin.

Gure Ormazabal et al. (2022b) lanean pibote itzulpen automatikoan oinarritutako parafrasi sistemak **matematikoki aztertzen** ditugu, eta frogatzen dugu inplizituki parafrasiak definitzen dituen funtzio matematiko bat ikasten dutela, hainbat gabezia dituena. Analisi honekin motibatuta, **ikasketa antagonikoan** oinarritutako metodo berri bat proposatzen dugu, corpus elebidunetatik parafrasi sistema bat ikasteko. Gainera, gure sistemak corpus elebakarretan oinarritutako sistemetan sarritan aurkitzen den **informazio-mugatzearen teknika barneratzen du**. Giza-ebaluazio eta ebaluazio automatikoaz gain, metodoarentzat berme matematikoak proposatu eta frogatzen ditugu.

1.5.3 Poesia-sorkuntza

Poesia-sorkuntza automatikoaren gakoa **egitura poetiko berezia** jarraitzen duen testua sortzean datza. [11] Lehen ahaleginek aldez aurretik definitutako erregelak erabiltzen zituzten beharrezko egitura jarraitzen duela bermatzeko. Gervás (2000) eta Gonçalo Oliveira et al. (2007) lanetan hurbilpen hau jarraitzen da, baina hitzak indibidualki ausaz lagintzen dituzte, eta ondorioz testuak ez du koherentzia globalik. **Erauzketan** oinarritutako hainbat hurbilpen ere proposatu izan dira, corpus handi batetik zatiak erauzi eta txantiloiak erabiliz nahi den egiturako poemak osatzen dituztenak Colton et al. (2012); Gonçalo Oliveira (2012); Gonçalo Oliveira et al. (2017).

[10] https://www.kaggle.com/c/quora-question-pairs

[11] Hemen egitura poetiko terminoa erabiltzen dugu poesia mota bakoitzak ezartzen duen egitura bereziari. Egitura berezi honen forma zehatza hizkuntzaren eta poesia motaren araberakoa izan ohi da. Adibidez, ingelesezko poesiak sarritan ez du errima eta silaba patroi zorrotzik zehazten, baina prosodia kontuan hartzen du. Euskarazko bertsolaritzak, berriz, errima eta silaba patroi zorrotzak ezartzen ditu, baina ez du prosodia kontuan hartzen.

Hala ere, parafrasi-sorkuntzan —eta hizkuntzaren prozesamenduko ikerkuntza-lerro ia guztietan— gertatu den bezala, gaur egungo poesia-sorkuntza sistema gehienak neurona-sareetan oinarritzen dira. neurona-sareak gai dira datu multzo handiak baliatuz kalitate handiko testu naturala sortzen ikasteko, baina poesia-sorkuntzara aplikatzeko muga argi bat dute: ezin da bermatu sortzen duen testuak egitura poetiko jakin bat jarraituko duenik. Erregela eta txantiloietan oinarritutako sistemek egitura jakin bateko —baina koherentzia global eskaseko— testua sortzeko mekanismo natural bat eskaintzen bazuten, sistema neuronaletan kontrako arazoa aurkitzen dugu. Lan-lerro batek egitura poetikoaren murriztapenak egoera-finituko automata (EFA) baten bidez adieraztea proposatzen du, eta ondoren EFAren bidez RNN sare baten irteera murriztea (Ghazvininejad et al., 2016, 2018; Hopkins and Kiela, 2017). Sistema hauek guztiek testu liriko edo poetikoa behar dute RNNa edo EFA entrenatzeko, eta gainera poema lerroak eskuinetik ezkerrera sortzen dituzte errimak errespetatu ahal izateko, ez baitute plangintza gaitasunik. Gainera, sortu nahi den egitura poetiko berri bakoitzeko dagokion EFA bat entrenatu behar dute sistema hauek.

EFA baten bidez egitura ezarri ordez, beste aukera bat egitura berezi hau datuetatik ikastea da. Lau et al. (2018) lanean RNN bat entrenatzen dute soneto corpus baten gainean, eta frogatzen dute sistemak sonetoen egitura errespetatzen ikasten duela. Van de Cruys (2020) lanean, berriz, corpus ez-poetiko arrunt bat erabiltzen dute sistema sortzeko, baina ez dute egitura poetiko zorrotz bat ezartzen. Aurreko kasuetan bezala, bi lan hauek eskuinetik ezkerrera sortzen dituzte poema lerroak, errimak sortzean plangintza faltaren arazoa arintzeko.

Arestian aipatutako lanek gaztelaniazko eta ingelesezko poesia-sorkuntzarako hurbilpen desberdinak proposatzen dituzte, baina denek bi muga argi dituzte komunean: alde batetik, corpus liriko edo poetikoak behar dituzte sistemak egitura poetikoa ikasi dezan, eta bestetik **ez dute plangintza gaitasunik**, eta ondorioz errimak errespetatu ahal izateko lerroak eskuinetik ezkerrera sortzen dituzte. Gure Ormazabal et al. (2022a) lanean bi arazo hauek ekiditen dituen euskara eta gaztelaniazko poema sortzaile bat proposatzen dugu, **hizkuntza-eredu kontrolagarri** batean oinarritzen dena. Gure hizkuntza-eredua bi pausutan entrenatzen dugu. Lehenik testu-corpus arrunt baten errima eta metrika patroi inplizituak etiketatzen ditugu, eta informazio hori kodetzen duten kontrol-kodeak txertatzen ditugu corpusean, **corpus aberastu** bat sortzeko. Ondoren, corpus aberastu hau erabiltzen dugu hizkuntza-eredu arrunt bat entrenatzeko. Ikasketan zehar ereduak HHA atazarako kontrol-kodeetako informazioa baliatzen ikasten duenez, posible da inferentzia garaian **egitura poetikoa deskribatzen duten kontrol-kodeak** ematea ereduari, eta horrela ereduak poesia sortzea. Gure eredua guztiz ez-gainbegiratua da, ez baitu testu liriko edo poetiko berezirik behar entrenamenduan, eta testua hizkuntza-eredu estandarrek bezala ezkerretik-eskuinera sortzeko gai da, kontrol-kodeei esker errimen plangintza egin baitezake.

Aipatzekoa da ere kontrol-kodeen ideia hizkuntzaren prozesamenduko arloan aldez aurretik erabili izan dela, poesia-sorkuntzatik kanpo. Keskar et al. (2019) lanean testu-corpus bat automatikoki erauzitako metadatuekin aberasten dute hizkuntza-eredu kontrolagarri bat entrenatzeko. Itzulpen automatiko arloan, kontrol-kodeak erabili izan dira itzulpenen formaltasuna (Sennrich et al., 2016), domeinua (Kobus et al., 2016) edo luzera (Lakew et al., 2019) kontrolatzeko. Schioppa et al. (2021) lanean kontrol-kode jarraituen erabilera aztertzen dute, aldi berean hainbat atributuren kontrolagarritasuna lortzeko helburuarekin.

Azkenik, gure lanean mota horretako poesia aztertu ez badugu ere, txinerazko poesia-sorkuntzaren inguruko lan ugari argitaratu izan dira (Wang et al., 2016; Zhang et al., 2017a; Liu et al., 2018; Yeh et al., 2019; Yi et al., 2018). Txinerazko poesiaren egitura euskaraz, ingelesez eta gaztelaniazko egiten den poesiarekiko nabari desberdina da; karaktere bakoitzak silaba bakarra deskribatzen du, eta metrikak murriztapen tonalak ezarri ohi ditu. Ondorioz, literaturan erabilitako hurbilpen eta teknikak ere desberdinak izaten dira.

1.5.4 Hizkuntza-ereduen egokitzea eta kutxa beltzak

Ereduen **egokitzeak** garrantzi handia hartu du hizkuntzaren prozesamenduko arloan, bereziki ezagutza transferentziak eta aurreikasketa-egokitze paradigmak azken hamarkadaren bukaeran gailendu zirenetik (ikus aurrekarietan 1.4.2 atala). Tradizionalki ataza edo erabilera bakoitzeko sistema bereiziak entrenatzen baziren, modu isolatuan, paradigma berri honetan **ezagutza orokorreko sistemak aurre-entrenatzen** dira, ondoren ataza bakoitzeko erabili ahalgo direnak. Aurrekarien atalean ikusi den bezala, eredu hauek entrenatutako adierazpen testuingurudunek lehenago erabiltzen ziren hitz-bektore ez-testuingurudunak ordezkatu zituzten. Eredu hauen inguruko lehen lan batzuetan eredu aurre-entrenatua ataza bakoitzeko entrenatzen zen ereduen ezaugarri bat baino ez zen (Peters et al., 2018) —hitz-bektore ez-testuingurudunekin gertatzen zen bezala— baina laster eredu aurre-entrenatuek **ataza bakoitzeko berariazko ereduak guztiz ordezkatu zituzten** (Howard and Ruder, 2018; Radford et al., 2018; Devlin et al., 2019). Lan hauek, ataza bakoitzeko eredu berri bat modu isolatuan entrenatu ordez, eredu aurre-entrenatua ataza bakoitzera egokitzea proposatzen zuten. Egokitze prozesu honen bidez, posible da eredu aurre-entrenatuaren ezagutza orokorra ataza berrira transferitzea. Hizkuntzaren prozesamenduan transferentzia-ikasketaren inguruko literatura-azterketa baterako, ikus (Ruder et al., 2019a).

Aurre-entrenatu eta egokitzearen paradigmaren egokitze zatia garrantzi handikoa da, azken ereduaren kalitatea baldintzatuko baitu. Lehen lanek **eredu osoa fintzea** —hau da, ataza bakoitzeko datuekin modu arruntean entrenatzen jarraitzea— proposatzen zuten (Devlin et al., 2019; Radford et al., 2018; Howard and Ruder, 2018). Baina, azken urteetan, ereduen eskala handitu ahala, ereduak fintzearen **konputazio kostua izugarri hazi da** (ikus aurrekarien 1.4.2 atala). Gauzak honela, ereduak baliabide konputazional urriagoekin findu ahal izateko metodoak proposatzen dituzten lan ugari argitaratu izan dira. Houlsby et al. (2019) lanean ereduan **parametro gehigarri gutxi batzuk gehitzea** proposatzen dute, eta bakarrik parametro horien balioak aldatzea egokitze garaian. Ildo beretik, Liu et al. (2022) lanean ereduaren sarrera geruzan bakarrik parametro gehigarriak entrenatzen dituzte, eta Hu et al. (2022) lanean eredu arkitekturaren rango-baxuko berparametrizazio bat baliatzen dute egokitze garaian entrenatutako parametro kopurua murrizteko. Teknika hauek guztiek nagusiki eredu bat findu ahal izateko behar den **memoria mugatzen dute**, GPU gutxiago izanda fintzea ahalbidetzen, baina ez dute konputazio kostua nabari murrizten, ereduaren entrenamenduan aurrera- eta atzera-pasaldia berdin-berdin egin behar baitira.

Erlazionatutako beste arazo bat **kutxa-beltz ereduen egokitzea da**. Egungo artearen-egoerako ereduak, besteak beste Claude, GPT-4 eta PaLM, denak APIen bidez bakarrik daude eskuragarri, eta ezin dira haien parametroak atzitu ez aldatu. Eredu hauek egokitze faserik gabe hainbat ataza betetzeko gai diren eredu anizkoitzak izan arren, sarritan domeinu edo ataza berri batean haien eraginkortasuna hobetzeko egokitu ahal izatea desiragarria da. Behar hau estaltzeko helburuarekin, kutxa-beltzen egokitzearen lan-lerroa ireki da azken urteetan. Shin et al. (2020) lanean ataza batean eraginkortasuna hobetzen duten *prompt*[12] berriak automatikoki osatzeko teknika bat proposatzen dute. Ildo beretik, ataza bakoitzerako *prompt* diseinu automatikoa aztertzen duten hainbat ahalegin egon dira (Sun et al., 2022; Zhang et al., 2023; Cheng et al., 2023). Metodo hauek kutxa-beltz ereduekin ondo ezkontzen diren arren, egokitzeko gaitasun mugatua dute, soilik *prompt* bat aldatzen ereduak ezin baitu dagoeneko aurreikasketan ikusi ez duen domeinu edo ataza berri bat ikasi.

Arestian aipatutako bi arazoak arintzeko, gure Ormazabal et al. (2023) lanean kutxa-beltz hizkuntza-ereduak konputazionalki eraginkorra den modu batean egokitzeko metodo bat aurkezten dugu. Zuzenean hizkuntza-eredua findu —edo *prompt* baten bidez kontrolatu— ordez, gure metodoak bi pausu jarraitzen ditu: lehenik eredu txiki bat fintzen du, domeinu edo ataza berrirako eredu aditu egokitu bat lortzeko. Ondoren, **eredu aditua kutxa-beltz ereduarekin konbinatzen da**, datuetatik ikasitako konbinaketa funtzio baten bidez. Modu honetan, kutxa-beltz ereduaren ezagutza orokorra baliatu dezakeen eredu egokitu bat lortu daiteke, haren parametroak atzitu —edo fintze osoak dakarren konputazio kostua pairatu— behar izan gabe.

Huang et al. (2023) lanean, gurearen aldi berean argitaratutakoa, antzeko ideia bat proposatzen dute. Kutxa-beltza eredu aditu findu batekin konbinatu ordez, corpus batetik erauzitako esaldien bidez definitzen den probabilitate banaketa batekin konbinatzen dute. Konbinaketa funtzioaren inguruko esplorazio mugatua egiten dute ere, eta kutxa-beltz

[12] *Prompt* bat hizkuntza-eredu batek ataza jakin bat bete dezan erabiltzen den testu-sarrera da. Adibidez, galdera erantzute ataza baterako, asistente moduan entrenatutako hizkuntza-eredu batentzako *prompt* posible bat honakoa izan zitekeen: *Galdera erantzute sistema bat zara. Mesedez erantzun hurrengo galdera:*

eredu oso txikiak aztertzen dituzte soilik.

2 Conclusions

In this thesis, we have made several advancements towards attribute controllability in NLP models. We have explored different techniques, and applied them to diverse tasks and model architectures. More concretely, we identify the following main **contributions**, related to the various research lines outlined in section 1.2:

- We develop a method to control the alignment of static word embeddings during their training, without any direct bilingual supervision. Our approach aims to combine the best of both kinds of existing bilingual word embedding techniques, joint methods and mapping methods. We draw on observations from our previous work, were we found joint methods—that learn embeddings for all languages directly in a joint space—to be superior, but more difficult to apply, than mapping based methods, due to their expensive bilingual supervision needs. We thus propose to first learn the word embeddings for a target language and keep then fixed, and to then learn the word embeddings for the source language while **controlling their alignment**, to make sure they stay aligned to the existing target language through the use of anchor words. We evaluate our method on bilingual lexicon induction and cross-lingual transfer, and show that it produces state-of-the-art unsupervised embeddings. This contribution is the result of the **L1.1** research line.

- We present a method to control the information content in the intermediate representation of a sequence-to-sequence encoder-decoder model, and leverage it to develop a paraphrase generation system. We focus on the case where the available training resource consists of bilingual parallel corpora, a common scenario in the field of paraphrasing. We conduct a novel mathematical analysis of round-trip translation, a widely used method for paraphrase generation, and find that its underlying paraphrasing similarity function has some issues. Namely, it is susceptible to *confounding translations*, where two sentences that share a possible translation may be considered paraphrases of each other even when they are not. Motivated by this analysis, we introduce an improved paraphrase similarity function, and implement a relaxation of it. Our implementation relies on the information bottleneck principle, that essentially **controls the maximum amount of information** that the intermediate representation can contain about the source sentence. By applying this principle to an encoder-decoder system trained for machine translation, the intermediate representations **learns the semantics** of the source sentence, while **forgetting surface level** details, and can thus be used for paraphrasing. We provide mathematical bounds and guarantees for our proposed method, and show that our approach naturally provides a mechanism to control the diversity-fidelity trade-off of the generated paraphrases. We conduct both automatic and human evaluation, showing our method outperforms a round-trip machine translation baseline. This contribution belongs to the **L1.2** research line.

- We present PoeLM, a meter- and rhyme-controllable language model for poetry generation. We achieve controllability of rhyme and meter through the use of **control codes** when training the language model. We follow a three stage process: (i) first, we annotate the existing meter and rhyme patterns in a regular natural language corpus, using an automatic annotator we develop for Spanish and Basque; (ii) second, we augment the corpus by interleaving control code sections, that describe the meter and rhyme patterns automatically extracted in the first phase; and (iii) third, we train a regular language model on the augmented corpus, so that the model will learn to **leverage the control codes** in order to better fulfill its next word prediction training task. At inference time, we flip this process, and feed the model with the **control codes corresponding to the desired poetic rhyme and meter**. Since the model has learned to respect these control codes during training, it generates poetic text when provided with them, even though it has not been trained on such text. While controllability through control codes had been explored before, we show for the first time that it can be succesfully applied to fine-grained and strict constraints such as imposing precise rhyme and meter. Additionally, ours is the first fully unsupervised neural poetry generator capable of respecting strict rhyme and meter patterns, and thanks to our use of control codes it does not need to resort to tricks common in previous systems, such as generating lines right-to-left. This contribution is the result of the **L1.3** research line.

- Finally, we introduce CombLM, a general approach for adapting black-box language models to new domains and tasks. While our previous work in this thesis focused on unsupervised methods for controlling individual attributes, here we switch focus to a supervised **general approach** for adapting models to new domains and tasks. We focus on the scenario where the **internal workings of the model are not available** and **fine-tuning the parameters is not an option**, which renders common adaptation approaches unusable, as they all depend on fine-tuning of or access to model weights. This scenario is becoming increasingly common, as regular fine-tuning of modern state-of-the-art language models is intractable for most researchers, either due to prohibitive computational cost, or black-box models accessible only through APIs. Our approach instead proposes to **leave the original model untouched**, and instead fine-tune a **small expert model** on the desired new domain or task. Then, the original black-box generalist model and the small expert are combined at the output probability level through a **learned combination** function, to obtain a new adapted model. We validate our approach on three domain adaptation tasks and one machine translation task, showing that the adapted model outperforms either of the original ones by a significant margin. CombLM is the product of the **L2** research line.

Regarding **publications**, the work done in this thesis has produced **four papers** accepted at top international conferences, two in ACL (Ormazabal et al., 2021, 2022b), and another two in EMNLP (Ormazabal et al., 2022a, 2023) (one of them accepted in Findings). Additionally, some of the work done during the development of PoeLM has contributed to a paper accepted in EPIA 2023 (Agirrezabal et al., 2023).

Additionally, we have made code and models publicly available for several of the developed methods.[1] [2]

The field of NLP has undergone rapid evolution during the development of this thesis, and we have seen popular approaches and architectures virtually disappear and be replaced by new ones. The architectures and tasks to which we have applied our methods have changed accordingly. One of the key factors in the success of modern models is their **scale**, with hundreds or thousands of accelerators used for months to train them. At the beginning of this thesis, the largest known NLP model was the then recently released GPT-3 model, trained with 175B parameters on 300B tokens, for a total approximate FLOP count of 3.110^{23}.[3] (Narayanan et al., 2021). While already enormous in scale, the GPT-3 model's scale pales in comparison with that of models currently openly available such as Falcon-180B, trained with 180B parameters on 3.5T tokens, for a total approximate FLOP count of 3.710^{24}, and current closed models are believed to be even larger in scale. Given this trend in scaling, we believe our work on efficient adaptation of black-box language models in the **L2** research line is particularly relevant, as it allows for adaptation either when not enough compute for fine-tuning a large model is available, as is the case for most non-industry actors, or when access to model internals is impossible. In the future, we would be interested in further exploring methods for adaptation of models when regular fine-tuning is not practical due to intractable scale.

[1] https://github.com/aitorormazabal/paraphrasingfromparallel

[2] https://github.com/aitorormazabal/poetry_generation

[3] FLOPs stand for total floating-point operations used during the training of a model, and a commonly used formula for estimating it is $C = 6ND$, where C is the total compute in FLOPs, N is the number of parameters, and D is the amount of training tokens

Glossary

arreta-gurutzatu *cross-attention*

arreta-mekanismo *attention mechanism*

atzera-pasaldi *backward pass*

aurre-entrenatu *pre-trained*

aurrera-pasaldi *forward pass*

auto-arreta *self-attention*

autogainbegiratua *self-supervised*

autoikasketa *self-learning*

autokodetzaile *autoencoder*

autokodetzaile bariazional *variational autoencoder*

autokodetze *autoencoding*

azpilaginketa *sub-sampling*

azpisekuentzia *subsequence*

fintze *fine-tuning*

gradiente desagerkorren arazoa *vanishing gradient problem*

hipotesi distribuzionala *distributional hypothesis*

hitz-bektore *word embedding*

hurrengo hitz aurresate *next word prediction*

ikasketa antagoniko *adversarial learning*

ikasketa-helburu *loss function*

informazio-mugatzearen teknika *information bottleneck method*

kodetzaile-deskodetzaile *encoder-decoder*

neurona-sare *neural network*

neurona-sare errepikari *recurrent neural network*

produktu eskalar egokitua *scaled dot product*

rango-baxuko berparametrizazioa *low-rank reparametrization*

transferentzia-ikasketa *transfer learning*

Bibliography

Manex Agirrezabal, Hugo Gonçalo Oliveira, and Aitor Ormazabal. 2023. Erato: Automatizing poetry evaluation.

Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Maitha Alhammadi, Mazzotta Daniele, Daniel Heslow, Julien Launay, Quentin Malartic, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo. 2023. The falcon series of language models: Towards open frontier models.

Rohan Anil, Andrew M. Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, Eric Chu, Jonathan H. Clark, Laurent El Shafey, Yanping Huang, Kathy Meier-Hellstern, Gaurav Mishra, Erica Moreira, Mark Omernick, Kevin Robinson, Sebastian Ruder, Yi Tay, Kefan Xiao, Yuanzhong Xu, Yujing Zhang, Gustavo Hernandez Abrego, Junwhan Ahn, Jacob Austin, Paul Barham, Jan Botha, James Bradbury, Siddhartha Brahma, Kevin Brooks, Michele Catasta, Yong Cheng, Colin Cherry, Christopher A. Choquette-Choo, Aakanksha Chowdhery, Clément Crepy, Shachi Dave, Mostafa Dehghani, Sunipa Dev, Jacob Devlin, Mark Díaz, Nan Du, Ethan Dyer, Vlad Feinberg, Fangxiaoyu Feng, Vlad Fienber, Markus Freitag, Xavier Garcia, Sebastian Gehrmann, Lucas Gonzalez, Guy Gur-Ari, Steven Hand, Hadi Hashemi, Le Hou, Joshua Howland, Andrea Hu, Jeffrey Hui, Jeremy Hurwitz, Michael Isard, Abe Ittycheriah, Matthew Jagielski, Wenhao Jia, Kathleen Kenealy, Maxim Krikun, Sneha Kudugunta, Chang Lan, Katherine Lee, Benjamin Lee, Eric Li, Music Li, Wei Li, YaGuang Li, Jian Li, Hyeontaek Lim, Hanzhao Lin, Zhongtao Liu, Frederick Liu, Marcello Maggioni, Aroma Mahendru, Joshua Maynez, Vedant Misra, Maysam Moussalem, Zachary Nado, John Nham, Eric Ni, Andrew Nystrom, Alicia Parrish, Marie Pellat, Martin Polacek, Alex Polozov, Reiner Pope, Siyuan Qiao, Emily Reif, Bryan Richter, Parker Riley, Alex Castro Ros, Aurko Roy, Brennan Saeta, Rajkumar Samuel, Renee Shelby, Ambrose Slone, Daniel Smilkov, David R. So, Daniel Sohn, Simon Tokumine, Dasha Valter, Vijay Vasudevan, Kiran Vodrahalli, Xuezhi Wang, Pidong Wang, Zirui Wang, Tao Wang, John Wieting, Yuhuai Wu, Kelvin Xu, Yunhan Xu, Linting Xue, Pengcheng Yin, Jiahui

Yu, Qiao Zhang, Steven Zheng, Ce Zheng, Weikang Zhou, Denny Zhou, Slav Petrov, and Yonghui Wu. 2023. Palm 2 technical report.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2016. Learning principled bilingual mappings of word embeddings while preserving monolingual invariance. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2289–2294, Austin, Texas. Association for Computational Linguistics.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2017. Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 451–462, Vancouver, Canada. Association for Computational Linguistics.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018a. Generalizing and improving bilingual word embedding mappings with a multi-step framework of linear transformations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1).

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018b. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 789–798, Melbourne, Australia. Association for Computational Linguistics.

Colin Bannard and Chris Callison-Burch. 2005. Paraphrasing with bilingual parallel corpora. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, page 597–604, USA. Association for Computational Linguistics.

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. 2014. Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 238–247, Baltimore, Maryland. Association for Computational Linguistics.

Regina Barzilay and Lillian Lee. 2003. Learning to paraphrase: An unsupervised approach using multiple-sequence alignment. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 16–23.

Regina Barzilay and Kathleen R. McKeown. 2001. Extracting paraphrases from a parallel corpus. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, ACL '01, page 50–57, USA. Association for Computational Linguistics.

Yoshua Bengio, Réjean Ducharme, and Pascal Vincent. 2000. A neural probabilistic language model. In *Advances in Neural Information Processing Systems*, volume 13. MIT Press.

Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew Dai, Rafal Jozefowicz, and Samy Bengio. 2016. Generating sentences from a continuous space. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pages 10–21, Berlin, Germany. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Hailong Cao, Tiejun Zhao, Shu Zhang, and Yao Meng. 2016. A distribution-based model to learn bilingual word embeddings. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1818–1827, Osaka, Japan. The COLING 2016 Organizing Committee.

Mingda Chen, Qingming Tang, Sam Wiseman, and Kevin Gimpel. 2019. Controllable paraphrase generation with a syntactic exemplar. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5972–5984, Florence, Italy. Association for Computational Linguistics.

Daixuan Cheng, Shaohan Huang, Junyu Bi, Yuefeng Zhan, Jianfeng Liu, Yujing Wang, Hao Sun, Furu Wei, Denvy Deng, and Qi Zhang. 2023. Uprise: Universal prompt retrieval for improving zero-shot evaluation.

Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, Doha, Qatar. Association for Computational Linguistics.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, page 160–167, New York, NY, USA. Association for Computing Machinery.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(76):2493–2537.

Simon Colton, Jacob Goodwin, and Tony Veale. 2012. Full-face poetry generation. In *International Conference on Computational Creativity 2012*, pages 95–102. University College Dublin.

Jocelyn Coulmance, Jean-Marc Marty, Guillaume Wenzek, and Amine Benhalloum. 2015. Trans-gram, fast cross-lingual word-embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1109–1113, Lisbon, Portugal. Association for Computational Linguistics.

Tim Van de Cruys. 2020. Automatic poetry generation from prosaic text. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2471–2480, Online. Association for Computational Linguistics.

Andrew M Dai and Quoc V Le. 2015. Semi-supervised sequence learning. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Georgiana Dinu, Angeliki Lazaridou, and Marco Baroni. 2015. Improving zero-shot learning by mitigating the hubness problem.

Bill Dolan, Chris Quirk, and Chris Brockett. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 350–356, Geneva, Switzerland. COLING.

Long Duong, Hiroshi Kanayama, Tengfei Ma, Steven Bird, and Trevor Cohn. 2016. Learning crosslingual word embeddings without bilingual corpora. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1285–1295, Austin, Texas. Association for Computational Linguistics.

Elozino Egonmwan and Yllias Chali. 2019. Transformer and seq2seq model for paraphrase generation. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 249–255, Hong Kong. Association for Computational Linguistics.

Manaal Faruqui and Chris Dyer. 2014. Improving vector space word representations using multilingual correlation. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 462–471, Gothenburg, Sweden. Association for Computational Linguistics.

J. R. Firth. 1957. A synopsis of linguistic theory 1930-55. 1952-59:1–32.

Pascale Fung. 1997. Finding terminology translations from non-parallel corpora. In *Fifth Workshop on Very Large Corpora.*

Ashwinkumar Ganesan, Francis Ferraro, and Tim Oates. 2021. Learning a reversible embedding mapping using bi-directional manifold alignment. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3132–3139, Online. Association for Computational Linguistics.

Nikesh Garera, Chris Callison-Burch, and David Yarowsky. 2009. Improving translation lexicon induction from monolingual corpora via dependency contexts and part-of-speech equivalences. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL-2009)*, pages 129–137, Boulder, Colorado. Association for Computational Linguistics.

Eric Gaussier, J.M. Renders, I. Matveeva, C. Goutte, and H. Dejean. 2004. A geometric view on bilingual lexicon extraction from comparable corpora. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 526–533, Barcelona, Spain.

Pablo Gervás. 2000. Wasp: Evaluation of different strategies for the automatic generation of spanish verse. In *Proceedings of the AISB-00 symposium on creative & cultural aspects of AI*, pages 93–100.

Marjan Ghazvininejad, Yejin Choi, and Kevin Knight. 2018. Neural poetry translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 67–71, New Orleans, Louisiana. Association for Computational Linguistics.

Marjan Ghazvininejad, Xing Shi, Yejin Choi, and Kevin Knight. 2016. Generating topical poetry. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1183–1191, Austin, Texas. Association for Computational Linguistics.

Hugo Gonçalo Oliveira. 2012. Poetryme: a versatile platform for poetry generation. *Computational Creativity, Concept Invention, and General Intelligence*, 1:21.

Hugo Gonçalo Oliveira, Amílcar Cardoso, and Francisco Pereira. 2007. Exploring different strategies for the automatic generation of song lyrics with tra-la-lyrics. In *Proceedings of 13th Portuguese Conference on Artificial Intelligence, EPIA*, pages 57–68.

Hugo Gonçalo Oliveira, Raquel Hervás, Alberto Díaz, and Pablo Gervás. 2017. Multilingual extension and evaluation of a poetry generator. *Natural Language Engineering*, 23(6):929–967.

Stephan Gouws, Yoshua Bengio, and Greg Corrado. 2015. Bilbowa: Fast bilingual distributed representations without word alignments. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 748–756, Lille, France. PMLR.

Stephan Gouws and Anders Søgaard. 2015. Simple task-specific bilingual word embeddings. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1386–1390, Denver, Colorado. Association for Computational Linguistics.

Edouard Grave, Armand Joulin, and Quentin Berthet. 2018. Unsupervised alignment of embeddings with wasserstein procrustes. *ArXiv*, abs/1805.11222.

Ankush Gupta, Arvind Agarwal, Prawaan Singh, and Piyush Rai. 2018. A deep generative framework for paraphrase generation. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press.

Zellig Harris. 1954. Distributional structure. *Word*, 10(23):146–162.

Bradley Hauer, Garrett Nicolai, and Grzegorz Kondrak. 2017. Bootstrapping unsupervised bilingual lexicon induction. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 619–624, Valencia, Spain. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Jack Hopkins and Douwe Kiela. 2017. Automatically generating rhythmic verse with neural networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 168–178, Vancouver, Canada. Association for Computational Linguistics.

Yedid Hoshen and Lior Wolf. 2018. Non-adversarial unsupervised word translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 469–478, Brussels, Belgium. Association for Computational Linguistics.

Tom Hosking and Mirella Lapata. 2021. Factorising meaning and form for intent-preserving paraphrasing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1405–1418, Online. Association for Computational Linguistics.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. Association for Computational Linguistics.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

J. Hu, Rachel Rudinger, Matt Post, and Benjamin Durme. 2019. Parabank: Monolingual bitext generation and sentential paraphrasing via lexically-constrained neural machine translation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33:6521–6528.

Eric Huang, Richard Socher, Christopher Manning, and Andrew Ng. 2012. Improving word representations via global context and multiple word prototypes. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 873–882, Jeju Island, Korea. Association for Computational Linguistics.

Kuan-Hao Huang and Kai-Wei Chang. 2021. Generating syntactically controlled paraphrases without using annotated parallel pairs. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1022–1033, Online. Association for Computational Linguistics.

Yangsibo Huang, Daogao Liu, Zexuan Zhong, Weijia Shi, and Yin Tat Lee. 2023. $k$nn-adapter: Efficient domain adaptation for black-box language models.

Nitish Shirish Keskar, Bryan McCann, Lav Varshney, Caiming Xiong, and Richard Socher. 2019. CTRL - A Conditional Transformer Language Model for Controllable Generation. *arXiv preprint arXiv:1909.05858*.

Alexandre Klementiev, Ivan Titov, and Binod Bhattarai. 2012. Inducing crosslingual distributed representations of words. In *Proceedings of COLING 2012*, pages 1459–1474, Mumbai, India. The COLING 2012 Organizing Committee.

Catherine Kobus, Josep Maria Crego, and Jean Senellart. 2016. Domain control for neural machine translation. *CoRR*, abs/1612.06140.

Tomáš Kočiský, Karl Moritz Hermann, and Phil Blunsom. 2014. Learning bilingual word representations by marginalizing alignments. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 224–229, Baltimore, Maryland. Association for Computational Linguistics.

Ashutosh Kumar, Kabir Ahuja, Raghuram Vadapalli, and Partha Talukdar. 2020. Syntax-Guided Controlled Generation of Paraphrases. *Transactions of the Association for Computational Linguistics*, 8:330–345.

Ashutosh Kumar, Satwik Bhattamishra, Manik Bhandari, and Partha Talukdar. 2019. Submodular optimization-based diverse paraphrasing and its effectiveness in data augmentation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3609–3619, Minneapolis, Minnesota. Association for Computational Linguistics.

Surafel Melaku Lakew, Mattia Di Gangi, and Marcello Federico. 2019. Controlling the output length of neural machine translation. In *Proceedings of the 16th International Conference on Spoken Language Translation*, Hong Kong. Association for Computational Linguistics.

Guillaume Lample, Alexis Conneau, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. In *International Conference on Learning Representations*.

Wuwei Lan, Siyu Qiu, Hua He, and Wei Xu. 2017. A continuously growing dataset of sentential paraphrases. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1224–1234, Copenhagen, Denmark. Association for Computational Linguistics.

Jey Han Lau, Trevor Cohn, Timothy Baldwin, Julian Brooke, and Adam Hammond. 2018. Deep-speare: A joint neural model of poetic language, meter and rhyme. In *Proceedings* *of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1948–1958, Melbourne, Australia. Association for Computational Linguistics.

S. Lauly, H. Larochelle, and Mitesh Khapra. 2014. An autoencoder approach to learning bilingual word representations. *Proc of NIPS*, pages 1853–1861.

Zichao Li, Xin Jiang, Lifeng Shang, and Hang Li. 2018. Paraphrase generation with deep reinforcement learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3865–3878, Brussels, Belgium. Association for Computational Linguistics.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *Computer Vision – ECCV 2014*, pages 740–755, Cham. Springer International Publishing.

Michael Littman, Susan Dumais, and Thomas Landauer. 1998. Automatic cross-language information retrieval using latent semantic indexing. *Automatic Cross-Language Information Retrieval Using Latent Semantic Indexing*.

Dayiheng Liu, Quan Guo, Wubo Li, and Jiancheng Lv. 2018. A multi-modal chinese poetry generation model. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. 2022. P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 61–68, Dublin, Ireland. Association for Computational Linguistics.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Bilingual word representations with monolingual quality in mind. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 151–159, Denver, Colorado. Association for Computational Linguistics.

Jonathan Mallinson, Rico Sennrich, and Mirella Lapata. 2017. Paraphrasing revisited with neural machine translation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 881–893, Valencia, Spain. Association for Computational Linguistics.

Kelly Marchisio, Neha Verma, Kevin Duh, and Philipp Koehn. 2022. IsoVec: Controlling the relative isomorphism of word embedding spaces. In *Proceedings of the 2022*

Conference on Empirical Methods in Natural Language Processing, pages 6019–6033, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Antonio Valerio Miceli Barone. 2016a. Towards cross-lingual distributed representations without parallel text trained with adversarial autoencoders. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 121–126. Association for Computational Linguistics.

Antonio Valerio Miceli Barone. 2016b. Towards cross-lingual distributed representations without parallel text trained with adversarial autoencoders. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 121–126, Berlin, Germany. Association for Computational Linguistics.

Sabrina Mielke, Zaid Alyafeai, Elizabeth Salesky, Colin Raffel, Manan Dey, Matthias Gallé, Arun Raja, Chenglei Si, Wilson Lee, Benoît Sagot, and Samson Tan. 2021. Between words and characters: A brief history of open-vocabulary modeling and tokenization in nlp.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *Proceedings of the Workshop track of the 2nd International Conference on Learning Representations (ICLR 2013).*

Tomás Mikolov, Quoc V. Le, and Ilya Sutskever. 2013b. Exploiting similarities among languages for machine translation. *CoRR*, abs/1309.4168.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013c. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26*, pages 3111–3119.

Ndapa Nakashole. 2018. NORMA: Neighborhood sensitive maps for multilingual word embeddings. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 512–522, Brussels, Belgium. Association for Computational Linguistics.

Ndapa Nakashole and Raphael Flauger. 2018. Characterizing departures from linearity in word translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 221–227, Melbourne, Australia. Association for Computational Linguistics.

Deepak Narayanan, Mohammad Shoeybi, Jared Casper, Patrick LeGresley, Mostofa Patwary, Vijay Anand Korthikanti, Dmitri Vainbrand, Prethvi Kashinkunti, Julie Bernauer, Bryan Catanzaro, Amar Phanishayee, and Matei Zaharia. 2021. Efficient large-scale language model training on gpu clusters using megatron-lm.

OpenAI. 2023. Gpt-4 technical report.

Aitor Ormazabal, Mikel Artetxe, and Eneko Agirre. 2023. CombLM: Adapting black-box language models through small fine-tuned models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2961–2974, Singapore. Association for Computational Linguistics.

Aitor Ormazabal, Mikel Artetxe, Manex Agirrezabal, Aitor Soroa, and Eneko Agirre. 2022a. PoeLM: A meter- and rhyme-controllable language model for unsupervised poetry generation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3655–3670, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Aitor Ormazabal, Mikel Artetxe, Gorka Labaka, Aitor Soroa, and Eneko Agirre. 2019. Analyzing the limitations of cross-lingual word embedding mappings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4990–4995, Florence, Italy. Association for Computational Linguistics.

Aitor Ormazabal, Mikel Artetxe, Aitor Soroa, Gorka Labaka, and Eneko Agirre. 2021. Beyond offline mapping: Learning cross-lingual word embeddings through context anchoring. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6479–6489, Online. Association for Computational Linguistics.

Aitor Ormazabal, Mikel Artetxe, Aitor Soroa, Gorka Labaka, and Eneko Agirre. 2022b. Principled paraphrase generation with parallel corpora. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1621–1638, Dublin, Ireland. Association for Computational Linguistics.

Bo Pang, Kevin Knight, and Daniel Marcu. 2003. Syntax-based alignment of multiple translations: Extracting paraphrases and generating new sentences. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 181–188.

Barun Patra, Joel Ruben Antony Moniz, Sarthak Garg, Matthew R. Gormley, and Graham Neubig. 2019. Bilingual lexicon induction with semi-supervision in non-isometric embedding spaces. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 184–193, Florence, Italy. Association for Computational Linguistics.

Yves Peirsman and Sebastian Padó. 2008. Semantic relations in bilingual lexicons. *ACM Trans. Speech Lang. Process.*, 8(2).

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Aaditya Prakash, Sadid A. Hasan, Kathy Lee, Vivek Datla, Ashequl Qadir, Joey Liu, and Oladimeji Farri. 2016. Neural paraphrase generation with stacked residual LSTM networks. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2923–2934, Osaka, Japan. The COLING 2016 Organizing Committee.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2023. Exploring the limits of transfer learning with a unified text-to-text transformer.

Reinhard Rapp. 1999. Automatic identification of word translations from unrelated English and German corpora. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 519–526, College Park, Maryland, USA. Association for Computational Linguistics.

Aurko Roy and David Grangier. 2019. Unsupervised paraphrasing without translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6033–6039, Florence, Italy. Association for Computational Linguistics.

Sebastian Ruder, Matthew E. Peters, Swabha Swayamdipta, and Thomas Wolf. 2019a. Transfer learning in natural language processing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials*, pages 15–18, Minneapolis, Minnesota. Association for Computational Linguistics.

Sebastian Ruder, Ivan Vulić, and Anders Søgaard. 2019b. A survey of cross-lingual word embedding models. *J. Artif. Int. Res.*, 65(1):569–630.

Sebastian Ruder, Ivan Vulić, and Anders Søgaard. 2019c. A survey of cross-lingual word embedding models. *J. Artif. Int. Res.*, 65(1):569–630.

Andrea Schioppa, David Vilar, Artem Sokolov, and Katja Filippova. 2021. Controlling machine translation for multiple attributes with additive interventions. In *Proceedings of* *the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6676–6696, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Controlling politeness in neural machine translation via side constraints. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 35–40, San Diego, California. Association for Computational Linguistics.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4222–4235, Online. Association for Computational Linguistics.

Yusuke Shinyama, Satoshi Sekine, and Kiyoshi Sudo. 2002. Automatic paraphrase acquisition from news articles. In *Proceedings of the Second International Conference on Human Language Technology Research*, HLT '02, page 313–318, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Anders Søgaard, Sebastian Ruder, and Ivan Vulić. 2018. On the limitations of unsupervised bilingual dictionary induction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 778–788, Melbourne, Australia. Association for Computational Linguistics.

Tianxiang Sun, Yunfan Shao, Hong Qian, Xuanjing Huang, and Xipeng Qiu. 2022. Black-box tuning for language-model-as-a-service.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Joseph Turian, Lev-Arie Ratinov, and Yoshua Bengio. 2010. Word representations: A simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 384–394, Uppsala, Sweden. Association for Computational Linguistics.

Peter Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*, 37.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 6000–6010.

Ivan Vulić, Goran Glavaš, Roi Reichart, and Anna Korhonen. 2019. Do we really need fully unsupervised cross-lingual embeddings? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4407–4418, Hong Kong, China. Association for Computational Linguistics.

Ivan Vulić and Anna Korhonen. 2016. On the role of seed lexicons in learning bilingual word embeddings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 247–257, Berlin, Germany. Association for Computational Linguistics.

Ivan Vulić and Marie-Francine Moens. 2015. Bilingual word embeddings from non-parallel document-aligned data applied to bilingual lexicon induction. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 719–725, Beijing, China. Association for Computational Linguistics.

Ivan Vulic and Marie-Francine Moens. 2016. Bilingual distributed word representations from document-aligned comparable data. *J. Artif. Int. Res.*, 55(1):953–994.

Zhe Wang, Wei He, Hua Wu, Haiyang Wu, Wei Li, Haifeng Wang, and Enhong Chen. 2016. Chinese poetry generation with planning based neural network. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1051–1060, Osaka, Japan. The COLING 2016 Organizing Committee.

Zirui Wang, Jiateng Xie, Ruochen Xu, Yiming Yang, Graham Neubig, and Jaime G. Carbonell. 2020. Cross-lingual alignment vs joint training: A comparative study and a simple unified framework. In *International Conference on Learning Representations*.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. Emergent abilities of large language models.

Joseph Weizenbaum. 1966. Eliza—a computer program for the study of natural language communication between man and machine. *Commun. ACM*, 9(1):36–45.

BigScience Workshop, :, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, Jonathan Tow, Alexander M. Rush, Stella Biderman, Albert Webson, Pawan Sasanka Ammanamanchi, Thomas Wang, Benoît Sagot, Niklas Muennighoff, Albert Villanova del Moral, Olatunji Ruwase, Rachel Bawden, Stas Bekman, Angelina McMillan-Major, Iz Beltagy, Huu Nguyen, Lucile Saulnier, Samson Tan, Pedro Ortiz Suarez, Victor Sanh, Hugo Laurençon, Yacine Jernite, Julien Launay, Margaret Mitchell, Colin Raffel, Aaron Gokaslan, Adi Simhi, Aitor Soroa, Alham Fikri Aji, Amit Alfassy, Anna Rogers, Ariel Kreisberg Nitzav, Canwen Xu, Chenghao Mou, Chris Emezue, Christopher Klamm, Colin Leong, Daniel van Strien, David Ifeoluwa Adelani, Dragomir Radev, Eduardo González Ponferrada, Efrat Levkovizh, Ethan Kim, Eyal Bar Natan, Francesco De Toni, Gérard Dupont, Germán Kruszewski, Giada Pistilli, Hady Elsahar, Hamza Benyamina, Hieu Tran, Ian Yu, Idris Abdulmumin, Isaac Johnson, Itziar Gonzalez-Dios, Javier de la Rosa, Jenny Chim, Jesse Dodge, Jian Zhu, Jonathan Chang, Jörg Frohberg, Joseph Tobing, Joydeep Bhattacharjee, Khalid Almubarak, Kimbo Chen, Kyle Lo, Leandro Von Werra, Leon Weber, Long Phan, Loubna Ben allal, Ludovic Tanguy, Manan Dey, Manuel Romero Muñoz, Maraim Masoud, María Grandury, Mario Šaško, Max Huang, Maximin Coavoux, Mayank Singh, Mike Tian-Jian Jiang, Minh Chien Vu, Mohammad A. Jauhar, Mustafa Ghaleb, Nishant Subramani, Nora Kassner, Nurulaqilla Khamis, Olivier Nguyen, Omar Espejel, Ona de Gibert, Paulo Villegas, Peter Henderson, Pierre Colombo, Priscilla Amuok, Quentin Lhoest, Rheza Harliman, Rishi Bommasani, Roberto Luis López, Rui Ribeiro, Salomey Osei, Sampo Pyysalo, Sebastian Nagel, Shamik Bose, Shamsuddeen Hassan Muhammad, Shanya Sharma, Shayne Longpre, Somaieh Nikpoor, Stanislav Silberberg, Suhas Pai, Sydney Zink, Tiago Timponi Torrent, Timo Schick, Tristan Thrush, Valentin Danchev, Vassilina Nikoulina, Veronika Laippala, Violette Lepercq, Vrinda Prabhu, Zaid Alyafeai, Zeerak Talat, Arun Raja, Benjamin Heinzerling, Chenglei Si, Davut Emre Taşar, Elizabeth Salesky, Sabrina J. Mielke, Wilson Y. Lee, Abheesht Sharma, Andrea Santilli, Antoine Chaffin, Arnaud Stiegler, Debajyoti Datta, Eliza Szczechla, Gunjan Chhablani, Han Wang, Harshit Pandey, Hendrik Strobelt, Jason Alan Fries, Jos Rozen, Leo Gao, Lintang Sutawika, M Saiful Bari, Maged S. Al-shaibani, Matteo Manica, Nihal Nayak, Ryan Teehan, Samuel Albanie, Sheng Shen, Srulik Ben-David, Stephen H. Bach, Taewoon Kim, Tali Bers, Thibault Fevry, Trishala Neeraj, Urmish Thakker, Vikas Raunak, Xiangru Tang, Zheng-Xin Yong, Zhiqing Sun, Shaked Brody, Yallow Uri, Hadar Tojarieh, Adam Roberts, Hyung Won Chung, Jaesung Tae, Jason Phang, Ofir Press, Conglong Li, Deepak Narayanan, Hatim Bourfoune, Jared Casper, Jeff Rasley, Max Ryabinin, Mayank Mishra, Minjia Zhang, Mohammad Shoeybi, Myriam Peyrounette, Nicolas Patry, Nouamane Tazi, Omar Sanseviero, Patrick von Platen, Pierre Cornette, Pierre François Lavallée, Rémi Lacroix, Samyam Rajbhandari, Sanchit Gandhi, Shaden Smith, Stéphane Requena, Suraj Patil, Tim Dettmers, Ahmed Baruwa, Amanpreet Singh, Anastasia Cheveleva, Anne-Laure Ligozat, Arjun Subramonian, Aurélie Névéol, Charles Lovering, Dan Garrette, Deepak Tunuguntla, Ehud Reiter, Ekaterina Taktasheva, Ekaterina Voloshina, Eli Bogdanov, Genta Indra Winata, Hailey Schoelkopf, Jan-Christoph Kalo, Jekaterina Novikova, Jessica Zosa Forde, Jordan Clive, Jungo Kasai, Ken Kawamura, Liam Hazan, Marine Carpuat, Miruna Clinciu, Najoung Kim, Newton Cheng, Oleg Serikov, Omer Antverg, Oskar van der Wal, Rui Zhang, Ruochen Zhang, Sebastian Gehrmann, Shachar Mirkin, Shani Pais, Tatiana Shavrina, Thomas Scialom, Tian Yun, Tomasz Limisiewicz, Verena Rieser, Vitaly Protasov, Vladislav Mikhailov, Yada Pruksachatkun, Yonatan Belinkov, Zachary Bamberger, Zdeněk Kasner, Alice Rueda, Amanda Pestana, Amir Feizpour, Ammar Khan, Amy Faranak, Ana Santos, Anthony Hevia, Antigona Unldreaj, Arash Aghagol, Arezoo Abdollahi, Aycha Tammour, Azadeh HajiHosseini, Bahareh Behroozi, Benjamin Ajibade, Bharat Saxena, Carlos Muñoz Ferrandis, Daniel McDuff, Danish Contractor, David Lansky, Davis David, Douwe Kiela, Duong A. Nguyen, Edward Tan, Emi Baylor, Ezinwanne Ozoani, Fatima Mirza, Frankline Ononiwu, Habib Rezanejad, Hessie Jones, Indrani Bhattacharya, Irene Solaiman, Irina Sedenko, Isar Nejadgholi, Jesse Passmore, Josh Seltzer, Julio Bonis Sanz, Livia Dutra, Mairon Samagaio, Maraim Elbadri, Margot Mieskes, Marissa Gerchick, Martha Akinlolu, Michael McKenna, Mike Qiu, Muhammed Ghauri, Mykola Burynok, Nafis Abrar, Nazneen Rajani, Nour Elkott, Nour Fahmy, Olanrewaju Samuel, Ran An, Rasmus Kromann, Ryan Hao, Samira Alizadeh, Sarmad Shubber, Silas Wang, Sourav Roy, Sylvain Viguier, Thanh Le, Tobi Oyebade, Trieu Le, Yoyo Yang, Zach Nguyen, Abhinav Ramesh Kashyap, Alfredo Palasciano, Alison Callahan, Anima Shukla, Antonio Miranda-Escalada, Ayush Singh, Benjamin Beilharz, Bo Wang, Caio Brito, Chenxi Zhou, Chirag Jain, Chuxin Xu, Clémentine Fourrier, Daniel León Periñán, Daniel Molano, Dian Yu, Enrique Manjavacas, Fabio Barth, Florian Fuhrimann, Gabriel Altay, Giyaseddin Bayrak, Gully Burns, Helena U. Vrabec, Imane Bello, Ishani Dash, Jihyun Kang, John Giorgi, Jonas Golde, Jose David Posada, Karthik Rangasai Sivaraman, Lokesh Bulchandani, Lu Liu, Luisa Shinzato, Madeleine Hahn de Bykhovetz, Maiko Takeuchi, Marc Pàmies, Maria A Castillo, Marianna Nezhurina, Mario Sänger, Matthias Samwald, Michael Cullan, Michael Weinberg, Michiel De Wolf, Mina Mihaljcic, Minna Liu, Moritz Freidank, Myungsun Kang, Natasha Seelam, Nathan Dahlberg, Nicholas Michio Broad, Nikolaus Muellner, Pascale Fung, Patrick Haller, Ramya Chandrasekhar, Renata Eisenberg, Robert Martin, Rodrigo Canalli, Rosaline Su, Ruisi Su, Samuel Cahyawijaya, Samuele Garda, Shlok S Deshmukh, Shubhanshu Mishra, Sid Kiblawi, Simon Ott, Sinee Sang-aroonsiri, Srishti Kumar, Stefan Schweter, Sushil Bharati, Tanmay Laud, Théo Gigant, Tomoya Kainuma, Wojciech Kusa, Yanis Labrak, Yash Shailesh Bajaj, Yash Venkatraman, Yifan Xu, Yingxin Xu, Yu Xu, Zhe Tan, Zhongli Xie, Zifan Ye, Mathilde Bras, Younes Belkada, and Thomas Wolf. 2023. Bloom: A 176b-parameter open-access multilingual language model.

Chao Xing, Dong Wang, Chao Liu, and Yiye Lin. 2015. Normalized word embedding and orthogonal transform for bilingual word translation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1006–1011, Denver, Colorado. Association for Computational Linguistics.

Dongjie Yang, Zhuosheng Zhang, and Hai Zhao. 2023. Learning better masking for better language model pre-training. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7255–7267, Toronto, Canada. Association for Computational Linguistics.

Pengcheng Yang, Fuli Luo, Shuangzhi Wu, Jingjing Xu, Dongdong Zhang, and Xu Sun. 2018. Learning unsupervised word mapping by maximizing mean discrepancy.

Wen-Chao Yeh, Yung-Chun Chang, Yu-Hsuan Li, and Wei-Chieh Chang. 2019. Rhyming knowledge-aware deep neural network for chinese poetry generation. In *2019 International Conference on Machine Learning and Cybernetics (ICMLC)*, pages 1–6.

Xiaoyuan Yi, Maosong Sun, Ruoyu Li, and Wenhao Li. 2018. Automatic poetry generation with mutual reinforcement learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3143–3153, Brussels, Belgium. Association for Computational Linguistics.

Biao Zhang, Barry Haddow, and Alexandra Birch. 2023. Prompting large language model for machine translation: A case study.

Jiyuan Zhang, Yang Feng, Dong Wang, Yang Wang, Andrew Abel, Shiyue Zhang, and Andi Zhang. 2017a. Flexible and creative Chinese poetry generation using neural memory. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1364–1373, Vancouver, Canada. Association for Computational Linguistics.

Meng Zhang, Yang Liu, Huanbo Luan, and Maosong Sun. 2017b. Adversarial training for unsupervised bilingual lexicon induction. In *Proceedings of the 55th Annual Meeting* *of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1959–1970, Vancouver, Canada. Association for Computational Linguistics.

Meng Zhang, Yang Liu, Huanbo Luan, and Maosong Sun. 2017c. Earth mover's distance minimization for unsupervised bilingual lexicon induction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1934–1945, Copenhagen, Denmark. Association for Computational Linguistics.

A
Appendix

This appendix includes a copy of the publications related to this thesis in the recommended reading order.

Beyond Offline Mapping: Learning Cross-lingual Word Embeddings through Context Anchoring

Aitor Ormazabal[1], **Mikel Artetxe**[2], **Aitor Soroa**[1], **Gorka Labaka**[1], **Eneko Agirre**[1]
[1]HiTZ Center, University of the Basque Country (UPV/EHU)
[2]Facebook AI Research
{aitor.ormazabal,a.soroa,gorka.labaka,e.agirre}@ehu.eus
artetxe@fb.com

Abstract

Recent research on cross-lingual word embeddings has been dominated by unsupervised mapping approaches that align monolingual embeddings. Such methods critically rely on those embeddings having a similar structure, but it was recently shown that the separate training in different languages causes departures from this assumption. In this paper, we propose an alternative approach that does not have this limitation, while requiring a weak seed dictionary (e.g., a list of identical words) as the only form of supervision. Rather than aligning two fixed embedding spaces, our method works by fixing the target language embeddings, and learning a new set of embeddings for the source language that are aligned with them. To that end, we use an extension of skip-gram that leverages translated context words as anchor points, and incorporates self-learning and iterative restarts to reduce the dependency on the initial dictionary. Our approach outperforms conventional mapping methods on bilingual lexicon induction, and obtains competitive results in the downstream XNLI task.

1 Introduction

Cross-lingual word embeddings (CLWEs) represent words from two or more languages in a shared space, so that semantically similar words in different languages are close to each other. Early work focused on jointly learning CLWEs in two languages, relying on a strong cross-lingual supervision in the form of parallel corpora (Luong et al., 2015; Gouws et al., 2015) or bilingual dictionaries (Gouws and Søgaard, 2015; Duong et al., 2016). However, these approaches were later superseded by offline mapping methods, which separately train word embeddings in different languages and align them in an unsupervised manner through self-learning (Artetxe et al., 2018; Hoshen and Wolf, 2018) or adversarial training (Zhang et al. 2017; Conneau et al., 2018a).

Despite the advantage of not requiring any parallel resources, mapping methods critically rely on the underlying embeddings having a similar structure, which is known as the *isometry assumption* Several authors have observed that this assumption does not generally hold, severely hindering the performance of these methods (Søgaard et al., 2018; Nakashole and Flauger, 2018; Patra et al., 2019). In later work, Ormazabal et al. (2019) showed that this issue arises from trying to align separately trained embeddings, as joint learning methods are not susceptible to it.

In this paper, we propose an alternative approach that does not have this limitation, but can still work without any parallel resources. The core idea of our method is to fix the target language embeddings, and learn aligned embeddings for the source language from scratch. This prevents structural mismatches that result from independently training embeddings in different languages, as the learning of the source embeddings is tailored to each particular set of target embeddings. For that purpose, we use an extension of skip-gram that leverages translated context words as anchor points. So as to translate the context words, we start with a weak initial dictionary, which is iteratively improved through self-learning, and we further incorporate a restarting procedure to make our method more robust Thanks to this, our approach can effectively work without any human-crafted bilingual resources, relying on simple heuristics (automatically generated lists of numerals or identical words) or an existing unsupervised mapping method to build the initial dictionary. Our experiments confirm the effectiveness of our approach, outperforming previous mapping methods on bilingual dictionary induction and obtaining competitive results on zero-shot cross-lingual transfer learning on XNLI.

2 Related work

Word embeddings. Embedding methods learn static word representations based on co-occurrence statistics from a corpus. Most approaches use two different matrices to represent the words and the contexts, which are known as the *input* and *output* vectors, respectively (Mikolov et al., 2013; Pennington et al., 2014; Bojanowski et al., 2017). The output vectors play an auxiliary role, being discarded after training. Our method takes advantage of this fact, leveraging translated output vectors as anchor points to learn cross-lingual embeddings. To that end, we build on the Skip-Gram with Negative Sampling (SGNS) algorithm (Mikolov et al., 2013), which trains a binary classifier to distinguish whether each output word co-occurs with the given input word in the training corpus or was instead sampled from a noise distribution.

Mapping CLWE methods. Offline mapping methods separately train word embeddings for each language, and then learn a mapping to align them into a shared space. Most of these methods align the embeddings through a linear map—often enforcing orthogonality constraints—and, as such, they rely on the assumption that the geometric structure of the separately learned embeddings is similar. This assumption has been shown to fail under unfavorable conditions, severely hindering the performance of these methods (Søgaard et al., 2018; Vulić et al., 2020). Existing attempts to mitigate this issue include learning non-linear maps in a latent space (Mohiuddin et al., 2020), employing maps that are only locally linear (Nakashole, 2018), or learning a separate map for each word (Glavaš and Vulić, 2020). However, all these methods are supervised, and have the same fundamental limitation of aligning a set of separately trained embeddings (Ormazabal et al., 2019).

Self-learning. While early mapping methods relied on a bilingual dictionary to learn the alignment, this requirement was alleviated thanks to self-learning, which iteratively re-induces the dictionary during training. This enabled learning CLWEs in a semi-supervised fashion starting from a weak initial dictionary (Artetxe et al., 2017), or in a completely unsupervised manner when combined with adversarial training (Conneau et al., 2018a) or initialization heuristics (Artetxe et al., 2018; Hoshen and Wolf, 2018). Our proposed method also incorporates a self-learning procedure, showing that this technique can also be effective with non-mapping methods.

Joint CLWE methods. Before the popularization of offline mapping, most CLWE methods extended monolingual embedding algorithms by either incorporating an explicit cross-lingual term in their learning objective, or directly replacing words with their translation equivalents in the training corpus. For that purpose, these methods relied on some form of cross-lingual supervision, ranging from bilingual dictionaries (Gouws and Søgaard 2015; Duong et al., 2016) to parallel or document-aligned corpora (Luong et al., 2015; Gouws et al. 2015; Vulić and Moens, 2016). More recently Lampie et al. (2018) reported positive results learning regular word embeddings over concatenated monolingual corpora in different languages, relying on identical words as anchor points. Wang et al. (2019) further improved this approach by applying a conventional mapping method afterwards As shown later in our experiments, our approach outperforms theirs by a large margin.

Freezing. Artetxe et al. (2020) showed that it is possible to transfer an English transformer to a new language by freezing all the inner parameters of the network and learning a new set of embeddings for the new language through masked language modeling. This works because the frozen transformer parameters constrain the resulting representations to be aligned with English. Similarly, our proposed approach uses frozen output vectors in the target language as anchor points to learn aligned embeddings in the source language.

3 Proposed method

Let $\mathbf{x}_i$ and $\tilde{\mathbf{x}}_i$ be the input and output vectors of the ith word in the source language, and $\mathbf{y}_j$ and $\tilde{\mathbf{y}}_j$ be their analogous in the target language.[1] In addition, let D be a bilingual dictionary, where $D(i) = j$ denotes that the ith word in the source language is translated as the jth word in the target language. Our approach first learns the target language embeddings $\{\mathbf{y}_i\}$ and $\{\tilde{\mathbf{y}}_i\}$ monolingually using regular SGNS. Having done that, we learn the source language embeddings $\{\mathbf{x}_i\}$ and $\{\tilde{\mathbf{x}}_i\}$ constraining them to be aligned with the target language embeddings according to the dictionary D. For that purpose, we propose an extension of

[1] Recall that $\{\tilde{\mathbf{x}}_i\}$ and $\{\tilde{\mathbf{y}}_j\}$ are auxiliary, and the goal is to learn aligned $\{\mathbf{x}_i\}$ and $\{\mathbf{y}_j\}$ (see §2).

Algorithm 1 Proposed method

```
Input: $D$ (dictionary), $C_{src}$ (src corpus), $C_{tgt}$ (tgt corpus)
Output: $\{\mathbf{x}_i\}$, $\{\mathbf{y}_i\}$ (aligned src and tgt embs)
Hparams: $T$ (updates), $R$ (restarts), $K$ (re-inductions)
$\{\mathbf{y}_i\}, \{\tilde{\mathbf{y}}_i\} \leftarrow \mathrm{SGNS}(C_{tgt})$    ▷ learn target embedings
for $r \leftarrow 1$ to $R$ do    ▷ iterative restart (§3.3)
    $\{\mathbf{x}_i\}, \{\tilde{\mathbf{x}}_i\} \leftarrow$ RANDOM_INIT()
    for $it \leftarrow 1$ to $T$ do
        $(w_i, w_j) \leftarrow$ NEXT_INSTANCE($C_{src}$)
        BACKPROP($\mathcal{L}(w_i, w_j)$)    ▷ core method (§3.1)
        if $it \mod (T/K) = 0$ then    ▷ self-learn (§3.2)
            $D \leftarrow$ REINDUCE($\{\mathbf{x}_i\}$, $\{\mathbf{y}_i\}$)
        end if
    end for
end for
```

SGNS that replaces the output vectors in the source language with their translation equivalents in the target language, which act as anchor points (§3.1). So as to make our method more robust to a weak initial dictionary, we incorporate a self-learning procedure that re-estimates the dictionary during training (§3.2), and perform iterative restarts (§3.3). Algorithm 1 summarizes our method.

3.1 SGNS with cross-lingual anchoring

Given a pair of words (w_i, w_j) co-occurring in the source language corpus, we define a generalized SGNS objective as follows:

$$\mathcal{L}(w_i, w_j) = \log \sigma \left(\mathbf{x}_{w_i} \cdot \mathrm{ctx}(w_j) \right) + \sum_{i=1}^{k} \mathbb{E}_{w_n \sim P_n(w)} \left[\log \sigma \left(-\mathbf{x}_{w_i} \cdot \mathrm{ctx}(w_n) \right) \right]$$

where k is the number of negative samples, $P_n(w)$ is the noise distribution, and $\mathrm{ctx}(w_t)$ is a function that returns the output vector to be used for w_t. In regular SGNS, this function would simply return the output vector of the corresponding word, so that $\mathrm{ctx}(w_t) = \tilde{\mathbf{x}}_{w_t}$. In contrast, our approach replaces it with its counterpart in the target language if w_t is in the dictionary:

$$\mathrm{ctx}(w_t) = \begin{cases} \tilde{\mathbf{y}}_{D(w_t)} & \text{if } w_t \in D \\ \tilde{\mathbf{x}}_{w_t} & \text{otherwise} \end{cases}$$

During training, the replaced vectors $\{\tilde{\mathbf{y}}_i\}$ are kept frozen, acting as anchor points so that the resulting embeddings $\{\mathbf{x}_i\}$ are aligned with their counterparts $\{\mathbf{y}_i\}$ in the target language.

3.2 Self-learning

As shown later in our experiments, the performance of our basic method is largely dependent on the quality of the bilingual dictionary itself. However, this is not different for conventional mapping methods, which also rely on a bilingual dictionary to align separately trained embeddings in different languages. So as to overcome this issue, modern mapping approaches rely on self-learning, which alternates between aligning the embeddings and re-inducing the dictionary in an iterative fashion (Artetxe et al., 2017).

We adopt a similar strategy, and re-induce the dictionary D a total of K times during training where K is a hyperparameter. To that end, we first obtain the translations for each source word using CSLS retrieval (Conneau et al., 2018a):

$$D(i) = \arg\max_j \mathrm{CSLS}(\mathbf{x}_i, \mathbf{y}_j)$$

Having done that, we discard all entries that do not satisfy the following cyclic consistency condition:[2]

$$i \in D \iff i = \arg\max_k \cos\left(\mathbf{x}_k, \mathbf{y}_{\arg\max_j \cos(\mathbf{x}_i, \mathbf{y}_j)} \right)$$

3.3 Iterative restarts

While self-learning is able to improve a weak initial dictionary throughout training, the method is still susceptible to poor local optima. This can be further exacerbated by the learning rate decay commonly used with SGNS, which makes it increasingly difficult to recover from a poor solution as training progresses. So as to overcome this issue we sequentially run the entire SGNS training R times, where R is a hyperparameter of the method We use the output from the previous run as the initial dictionary, but all the other parameters are reset and the full training process is run from scratch.

4 Experimental setup

We next describe the systems explored in our experiments (§4.1), the data and procedure used to train them (§4.2), and the evaluation tasks (§4.3).

4.1 Systems

We compare 3 model families in our experiments:

Offline mapping. This approach learns monolingual embeddings in each of the languages separately, which are then mapped into a common space

[2] We define our cyclic consistency condition over cosine similarity, which we found to be more restrictive than CSLS (in that it discards more entries) and work better in our preliminary experiments.

	en	de	es	fr	fi	ru	zh
Tokens	2,390	860	601	724	91	498	234
Sentences	101	42	22	28	6	25	10

Table 1: Size of the training corpora (millions).

	de-en	es-en	fr-en	fi-en	ru-en	zh-en
Identical	44.8	57.6	63.8	37.7	4.3	3.3
Numerals	1.4	1.6	1.6	2.4	1.1	0.2
Mapping	53.3	67.3	69.7	22.3	28.2	17.1

Table 2: Size of the initial dictionaries (thousands).

through a linear transformation. We experiment with 3 popular methods from the literature: MUSE (Conneau et al., 2018a), ICP (Hoshen and Wolf, 2018) and VecMap (Artetxe et al., 2018). We use the original implementation of each method in their unsupervised mode with default hyperparameters.

Joint learning + offline mapping. This approach jointly learns word embeddings for two languages over their concatenated monolingual corpora, where identical words act as anchor points (Lample et al., 2018). Having done that, the vocabulary is partitioned into one shared and two language specific subsets, which are further aligned through an offline mapping method (Wang et al., 2019). We use the joint_align implementation from the authors with default hyperparameters, which relies on fastText for the joint learning step and MUSE for the mapping step.[3]

Cross-lingual anchoring. Our proposed method, described in Section 3. We explore 3 alternatives to obtain the initial dictionary: **(i) identical words**, where $D_i = j$ if the ith source word and the jth target word are identically spelled, **(ii) numerals**, a subset of the former where identical words are further restricted to be sequences of digits, and **(iii) unsupervised mapping**, where we use the baseline VecMap system described above to induce the initial dictionary.[4] The first two variants make assumptions on the writing system of different languages, which is usually regarded as a weak form of supervision (Artetxe et al., 2017; Søgaard et al., 2018), whereas the latter is strictly unsupervised, yet dependant on an additional system from a different family.

4.2 Data and training details

We learn CLWEs between English and six other languages: German, Spanish, French, Finnish, Russian and Chinese. Following common practice, we use Wikipedia as our training corpus,[5] which we preprocessed using standard Moses scripts, and restrict our vocabulary to the most frequent 200K tokens per language. In the case of Chinese, word segmentation was done using the Stanford Segmenter. Table 1 summarizes the statistics of the resulting corpora, while Table 2 reports the sizes of the initial dictionaries derived from it for our proposed method.

For joint_align, we directly run the official implementation over our tokenized corpus as described above. All the other systems take monolingual embeddings as input, which we learn using the SGNS implementation in word2vec.[6] For our proposed method, we set English as the target language, fix the corresponding monolingual embeddings, and learn aligned embeddings in the source language using our extension of SGNS (§3).[7] We set the number of restarts R to 3, the number of reinductions per restart K to 50, and the number of epochs to $10\frac{\#\text{trg sents}}{\#\text{src sents}}$, which makes sure that the source language gets a similar number of updates to the 10 epochs done for English.[8] For all the other hyperparameters, we use the same values as for the monolingual embeddings. We made all of our development decisions based on preliminary experiments on English-Finnish, without any systematic hyperparameter exploration. Our implementation runs on CPU, except for the dictionary reinduction steps, which run on a single GPU for around one

[3] The original implementation only supports the supervised mode with RCSLS mapping, so we modified it to use MUSE in the unsupervised setting as described in the original paper.

[4] We use CSLS retrieval and apply the cyclic consistency restriction as described in Section 3.2.

[5] We extracted the corpus from the February 2019 dump using the WikiExtractor tool.

[6] We use 10 negative samples, a sub-sampling threshold of 1e-5, 300 dimensions, and 10 epochs. Note that joint_align also learns 300-dimensional vectors, but runs fastText with default hyperparameters under the hood.

[7] In our preliminary experiments, we observed our proposed method to be quite sensitive to which language is the source and which one is the target. We find the language with the largest corpus to perform best as the target, presumably because the corresponding monolingual embeddings are better estimated, so it is more appropriate to fix them and learn aligned embeddings for the other language. Following this observation, we set English as the target language for all pairs as it is the language with the largest corpus.

[8] For a fair comparison, we also tried using the same number of epochs for the baseline systems, but this performed worse than the reported numbers with 10 epochs.

	de-en		es-en		fr-en		fi-en		ru-en		zh-en		avg
	→	←	→	←	→	←	→	←	→	←	→	←	
OFFLINE MAPPING													
MUSE (Conneau et al., 2018a)	72.9	74.8	83.5	83.0	81.7	82.3	0.3*	0.3*	0.0*	0.3*	39.5	30.9	45.8
ICP (Hoshen and Wolf, 2018)	73.9	75.1	82.5	83.2	80.5	82.3	0.3*	0.3*	59.5	46.1	0.1*	2.8*	48.9
VecMap (Artetxe et al., 2018)	74.5	76.6	83.5	83.3	82.7	83.0	61.9	45.1	**65.7**	49.0	42.4	33.4	65.1
JOINT LEARNING + OFFLINE MAPPING													
Joint_Align (Wang et al., 2019)	70.7	68.7	71.9	69.6	79.2	78.0	33.1	29.1	31.3	25.1	3.6*	18.4	48.2
CROSS-LINGUAL ANCHORING													
Ours (identical init)	76.7	77.9	**86.5**	84.1	**85.0**	84.8	63.3	51.3	65.3	**51.6**	42.1	**38.9**	67.3
Ours (numeral init)	**76.9**	77.7	86.3	84.1	**85.0**	**84.9**	63.6	50.6	64.9	51.4	1.4*	4.9*	61.0
Ours (mapping init)	76.8	**78.1**	86.3	**84.2**	84.9	**84.9**	**64.2**	**51.5**	**65.7**	51.5	**42.5**	38.8	**67.5**

Table 3: Main BLI results on the MUSE dataset (P@1). Asterisks denote divergence ($< 5\%$ P@1).

hour in total.

4.3 Evaluation tasks

As described next, we evaluate our method on two tasks: Bilingual Lexicon Induction (BLI) and Cross-lingual Natural Language Inference (XNLI).

BLI. Following common practice, we induce a bilingual dictionary through CSLS retrieval (Conneau et al., 2018a) for each set of cross-lingual embeddings, and evaluate the precision at 1 (P@1) with respect to the gold standard test dictionary from the MUSE dataset (Conneau et al., 2018a). For the few out-of-vocabulary source words, we revert to copying as a back-off strategy,[9] so our reported numbers are directly comparable to prior work in terms of coverage.

XNLI. We train an English natural language inference model on MultiNLI (Williams et al., 2018), and evaluate the zero-shot cross-lingual transfer performance on the XNLI test set (Conneau et al., 2018b) for the subset of our languages covered by it. To that end, we follow Glavaš et al. (2019) and train an Enhanced Sequential Inference Model (ESIM) on top of our original English embeddings, which are kept frozen during training. At test time, we transfer into the rest of the languages by plugging in the corresponding aligned embeddings. Note that we use the exact same English model for our proposed method and the baseline MUSE and ICP systems,[10] which only differ in the set of aligned embeddings used for cross-lingual transfer. In contrast, VecMap and joint_align also manipulate the target English embeddings, which would require training a separate model for each language pair so we decide to exclude them from this set of experiments.[11]

5 Results

We next discuss our main results on BLI (§5.1) and XNLI (§5.2), followed by our ablation study (§5.3) and error analysis (§5.4) on BLI.

5.1 BLI

Table 3 comprises our main BLI results. We observe that our method obtains the best results in all directions (matched by VecMap in Russian-English), outperforming the strongest baseline by 2.4 points on average for the mapping based initialization. Our improvements are more pronounced in the backward direction (3.1 points on average) but still substantial in the forward direction (1.7 points on average).

It is worth noting that some systems fail to converge to a good solution for the most challenging language pairs. This includes our proposed method in the case of Chinese-English when using the numeral-based initialization, which we attribute to the smaller size of the initial dictionary (only 244 entries, see Table 2). Other than that, we observe that our approach obtains very similar results regardless of the initial dictionary. Quite remarkably

[9] This has a negligible impact in practice, as it accounts for less than 1.4% of the test cases. Moreover, all of our systems use the same underlying vocabulary, so they are affected in the exact same way.

[10] This is possible because they all fix the target language embeddings (English in this case) and align the embeddings in the source language with them, either through mapping (MUSE, ICP) or learning from scratch (ours).

[11] In addition to the computational overhead, having separate models introduces some variance, while our comparison is more direct.

	de-en		es-en		fr-en		ru-en		avg
	→	←	→	←	→	←	→	←	
Conneau et al. (2018a)	72.2	74.0	83.3	81.7	82.1	82.3	59.1	44.0	72.3
Hoshen and Wolf (2018)	73.0	74.7	84.1	82.1	82.9	82.3	61.8	47.5	73.6
Grave et al. (2018)	73.3	75.4	84.1	82.8	82.9	82.6	59.1	43.7	73.0
Alvarez-Melis and Jaakkola (2018)	72.8	71.9	80.4	81.7	78.9	81.3	43.7	45.1	69.5
Yang et al. (2018)	70.3	71.5	79.3	79.9	78.9	78.4	-	-	-
Mukherjee et al. (2018)	-	-	79.2	**84.5**	-	-	-	-	-
Alvarez-Melis et al. (2018)	71.1	73.8	81.8	81.3	81.6	82.9	55.4	41.7	71.2
Xu et al. (2018)	67.0	69.3	77.8	79.5	75.5	77.9	-	-	-
Wang et al. (2019)	72.2	74.2	84.2	81.4	83.6	82.8	58.3	45.0	72.7
Zhou et al. (2019)	74.4	77.2	84.9	82.8	83.5	83.1	63.6	49.2	74.8
Li et al. (2020)	74.3	75.3	84.6	82.4	83.7	82.6	-	-	-
Ours (mapping init)	**76.8**	**78.1**	**86.3**	84.2	**84.9**	**84.9**	**65.7**	**51.5**	**76.6**

Table 4: BLI results on MUSE dataset in comparison with prior published results (P@1). All systems are fully unsupervised (except that of Zhou et al. (2019), which uses identical words as a seed dictionary), and use SGNS embeddings trained on Wikipedia.

	en	de	es	fr	ru	zh
MUSE	**73.9**	65.0	68.1	**67.9**	39.1*	**61.4**
ICP	**73.9**	62.2	64.2	65.7	59.4	36.1*
Ours (identical init)	**73.9**	65.0	**68.7**	67.1	**63.5**	49.8
Ours (numeral init)	**73.9**	65.0	68.6	67.1	63.3	34.9*
Ours (mapping init)	**73.9**	**65.1**	68.6	67.0	**63.5**	49.4

Table 5: XNLI results (accuracy). Asterisks denote divergence ($< 5\%$ P@1 in BLI).

Basic method (identical init)	53.9
+ *self-learning*	66.9
+ *iterative restarts*	67.3
Basic method (numeral init)	2.6
+ *self-learning*	53.9
+ *iterative restarts*	61.0
Basic method (mapping init)	67.5
+ *self-learning*	67.5
+ *iterative restarts*	67.5

Table 6: Ablation results on BLI (average P@1)

the variant using VecMap for initialization (*mapping init*) is substantially stronger than VecMap itself despite not using any additional training signal.

So as to put our results into perspective, Table 4 compares them to previous numbers reported in the literature. Note that the numbers are comparable in terms of coverage and all systems use Wikipedia as the training corpus, although they might differ on the specific dump used and the preprocessing details.[12] As it can be seen, our approach obtains the best results by a substantial margin.[13]

5.2 XNLI

We report our XNLI results in Table 5. We observe that our method is competitive with the baseline mapping systems, achieving the best results on 3 out of the 5 transfer languages by a small margin Nevertheless, it significantly lags behind MUSE on Chinese, even if the exact same set of cross-lingual embeddings performed better than MUSE at BLI. While striking, similar discrepancies between BLI and XNLI performance where also observed in previous studies (Glavaš et al., 2019) Finally, we observe that the initial dictionary has a negligible impact in the performance of our proposed method, which supports the idea that our approach converges to a similar solution given any reasonable initialization.

5.3 Ablation study

So as to understand the role of self-learning and the iterative restarts in our approach, we perform an ablation study and report our results in Table 6 We observe that the contribution of these components is greatly dependant on the initial dictionary For the numeral initialization, the basic method works poorly, and both extensions bring large improvements. In contrast, the identical initialization

[12] In particular, most mapping methods use the official Wikipedia embeddings from fastText. Unfortunately, the pre-processed corpus used to train these embeddings is not public, so works that explore other approaches, like ours, need to use their own pre-processed copy of Wikipedia.

[13] Artetxe et al. (2019) report even stronger results based on unsupervised machine translation instead of direct retrieval with CLWEs. Note, however, that their method still relies on cross-lingual embeddings to build the underlying phrase-table, so our improvements should be largely orthogonal to theirs.

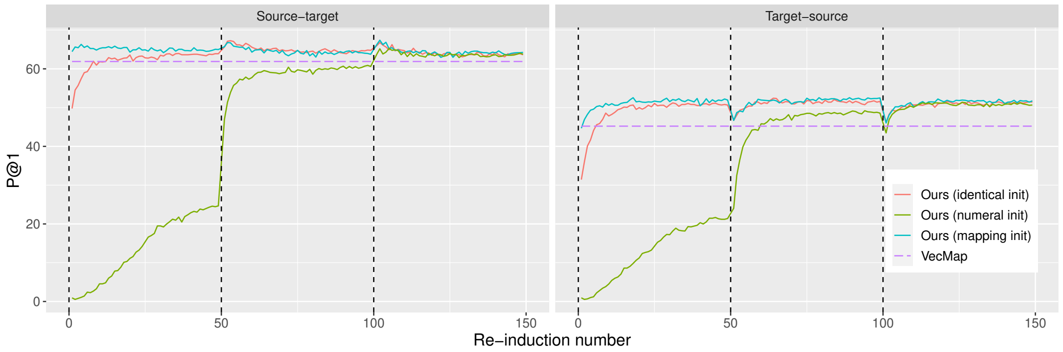


Figure 1: Finnish-English learning curves (BLI P@1). The iterative restarts happen at the vertical lines.

does not benefit from iterative restarts, but self-learning still plays a major role. In the case of the mapping-based initialization, the basic method is already very competitive. This suggests that both the self-learning and the iterative restarts are helpful to make the method more robust to a weak initialization, and have a minor impact otherwise.

In order to better understand the underlying learning dynamics, we analyze the learning curves for Finnish-English in Figure 1. We observe that, when the initial dictionary is strong, our method surpasses the baseline and stabilizes early. In contrast, convergence is much slower when using the weak numeral-based initialization, and the iterative restarts are critical to escape poor local optima.

5.4 Error analysis

So as to better understand where our improvements in BLI are coming from, we perform an error analysis on the Spanish-English direction. To that end, we manually inspect the 69 instances for which our method (with mapping-based initialization) produced a correct translation while VecMap failed according to the gold standard, as well as the 26 instances for which the opposite was true. We then categorize these errors into several types, which are summarized in Table 7.

We observe that, in 52.6% of the 95 analyzed instances, the translation produced by our method is identical to the source word, while this percentage goes down to 4.2% for VecMap. This tendency of our approach to copy its input is striking, as the model has no notion about the words being identically spelled.[14] A large portion of these cases correspond to named entities, where copying is the right behavior, while VecMap outputs a different proper noun. There are also some instances where the input word is in the target language,[15] which can be considered an artifact of the dataset, but copying also seems the most reasonable behavior in these cases. Finally, there are also a few cases where the input word is present in the target vocabulary, which is selected by our method and counted as an error. Once again, we consider these to be an artifact of the dataset, as copying seems a reasonable choice if the input word is considered to be part of the target language vocabulary. The remaining cases where neither method copies mostly correspond to common errors, where one of the systems (most often VecMap) outputs a semantically related but incorrect translation. However, there are also a few instances where both translations are correct, but one of them is missing in the gold standard.

With the aim to understand the impact of identical words in our original results, we re-evaluated the systems using a filtered version of the MUSE gold standard dictionaries, where we removed all source words that were included in the set of candidate translations. In order to be fair, we filtered out identical words from the output of the system reverting to the second highest-ranked translation whenever the first one is identical to the source word. The results for the strongest system in each family are shown in Table 8. Even if the margin of improvement is reduced compared to Table 3, the best results are still obtained by our proposed method, bringing an average improvement

[14] The variants of our system with identical or numeral initialization do indirectly see this signal, but the one analyzed here is initialized with the VecMap mapping.

[15] English words will often appear in other languages as part of named entities (e.g., “pink” as part of “Pink Floyd”), which explains the presence of such words in the Spanish vocabulary.

			Examples		
Gold standard	Type	Cases	Source	VecMap	Ours
Ours right – VecMap wrong	Common errors	30.5%	derrotas campeona	victories medalist	defeats champion
	Named entity, ours copies	21.1%	philadelphia susana	pittsburgh beatriz	philadelphia susana
	EN word in ES vocab, ours copies	15.8%	pink space	tangerine sci	pink space
	Gap in gold standard	5.3%	adecuada marquesa	appropriate marchioness	adequate marquise
VecMap right – Ours wrong	Common errors	15.8%	conservadores noveno	conservatives ninth	liberals tenth
	ES word in EN vocab, ours copies	7.4%	calzada cantera	roadway quarry	calzada cantera
	Gap in gold standard	4.2%	ferroviario situados	railway situated	rail positioned

Table 7: BLI error analysis on Spanish-English. See Section 5.4 for details.

	de-en		es-en		fr-en		fi-en		ru-en		zh-en		avg
	→	←	→	←	→	←	→	←	→	←	→	←	
VecMap (Artetxe et al., 2018)	68.3	70.2	85.1	79.4	80.8	78.1	**58.4**	38.9	**66.1**	48.6	45.0	34.5	62.8
Joint_Align (Wang et al., 2019)	57.0	53.3	63.0	57.4	70.2	64.4	4.0*	0.7*	31.3	22.4	3.5*	0.9*	35.7
Ours (identical init)	68.9	72.2	**86.0**	80.7	81.5	80.0	54.0	41.0	65.7	50.9	44.6	**38.1**	63.6
Ours (mapping init)	**68.9**	**72.3**	85.8	**80.8**	**81.4**	**80.2**	55.4	**41.6**	**66.1**	**51.0**	**45.1**	37.9	**63.9**

Table 8: BLI results on MUSE with identical words removed (P@1). Asterisks denote divergence ($< 5\%$ P@1).

of 1.1 points. It is also worth noting that joint_align, which shares a portion of the vocabulary for both languages (and will thus translate all words in the shared vocabulary identically), suffers a large drop in performance. This highlights the importance of accompanying quantitative BLI evaluation with an error analysis as urged by previous studies (Kementchedjhieva et al., 2019).

6 Conclusions and future work

Our approach for learning CLWEs addresses the main limitations of both offline mapping and joint learning methods. Different from mapping approaches, it does not suffer from structural mismatches arising from independently training embeddings in different languages, as it works by constraining the learning of the source embeddings so they are aligned with the target ones. At the same time, unlike previous joint methods, our system can work without any parallel resources, relying on numerals, identical words or an existing mapping method for the initialization. We achieve this by combining cross-lingual anchoring with self-learning and iterative restarts. While recent research on CLWEs has been dominated by mapping approaches, our work shows that the fundamental techniques that popularized these methods (e.g. the use of self-learning to relax the need for cross-lingual supervision) can also be effective beyond this paradigm.

Despite its simplicity, our experiments on BLI show the superiority of our method when compared to previous mapping systems. We complement these results with additional experiments on a downstream task, where our method obtains competitive results, as well as an ablation study and a systematic error analysis. We identify a striking tendency of our method to translate words identically, even if it has no notion of the words being identically spelled. Thanks to this, our method is particularly strong at translating named entities but we show that our improvements are not limited to this phenomenon. These insights confirm the value of accompanying quantitative results on BLI with qualitative evaluation (Kementchedjhieva et al., 2019) and/or other tasks (Glavaš et al., 2019)

In the future, we would like to further explore CLWE methods that go beyond the currently dominant mapping paradigm. In particular, we would like to remove the requirement of a seed dictionary altogether by using adversarial learning, and explore more elaborated context translation and dictionary re-induction schemes.

Acknowledgments

Aitor Ormazabal, Aitor Soroa, Gorka Labaka and Eneko Agirre were supported by the Basque Government (excellence research group IT1343-19 and DeepText project KK-2020/00088), project BigKnowledge (*Ayudas Fundación BBVA a equipos de investigación científica 2018*) and the Spanish MINECO (project DOMINO PGC2018-102041-B-I00 MCIU/AEI/FEDER, UE). Aitor Ormazabal was supported by a doctoral grant from the Spanish MECD.

References

David Alvarez-Melis and Tommi Jaakkola. 2018. Gromov-wasserstein alignment of word embedding spaces. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1881–1890, Brussels, Belgium. Association for Computational Linguistics.

David Alvarez-Melis, Stefanie Jegelka, and Tommi S Jaakkola. 2018. Towards optimal transport with global invariances. *arXiv preprint arXiv:1806.09277*.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2017. Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 451–462, Vancouver, Canada. Association for Computational Linguistics.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 789–798. Association for Computational Linguistics.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2019. Bilingual lexicon induction through unsupervised machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5002–5007, Florence, Italy. Association for Computational Linguistics.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637, Online. Association for Computational Linguistics.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018a. Word translation without parallel data. In *Proceedings of the 6th International Conference on Learning Representations (ICLR 2018)*.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. 2018b. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485, Brussels, Belgium. Association for Computational Linguistics.

Long Duong, Hiroshi Kanayama, Tengfei Ma, Steven Bird, and Trevor Cohn. 2016. Learning crosslingual word embeddings without bilingual corpora. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1285–1295, Austin, Texas. Association for Computational Linguistics.

Goran Glavaš, Robert Litschko, Sebastian Ruder, and Ivan Vulić. 2019. How to (properly) evaluate cross-lingual word embeddings: On strong baselines, comparative analyses, and some misconceptions. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 710–721, Florence, Italy. Association for Computational Linguistics.

Goran Glavaš and Ivan Vulić. 2020. Non-linear instance-based cross-lingual mapping for non-isomorphic embedding spaces. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7548–7555, Online. Association for Computational Linguistics.

Stephan Gouws, Yoshua Bengio, and Greg Corrado. 2015. BilBOWA: Fast bilingual distributed representations without word alignments. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 748–756.

Stephan Gouws and Anders Søgaard. 2015. Simple task-specific bilingual word embeddings. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1386–1390, Denver, Colorado. Association for Computational Linguistics.

Edouard Grave, Armand Joulin, and Quentin Berthet. 2018. Unsupervised alignment of embeddings with wasserstein procrustes. *arXiv preprint arXiv:1805.11222*.

Yedid Hoshen and Lior Wolf. 2018. Non-adversarial unsupervised word translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 469–478, Brussels, Belgium. Association for Computational Linguistics.

Yova Kementchedjhieva, Mareike Hartmann, and Anders Søgaard. 2019. Lost in evaluation: Misleading benchmarks for bilingual dictionary induction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3336–3341, Hong Kong, China. Association for Computational Linguistics.

Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Phrase-based & neural unsupervised machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5039–5049, Brussels, Belgium. Association for Computational Linguistics.

Yanyang Li, Yingfeng Luo, Ye Lin, Quan Du, Huizhen Wang, Shujian Huang, Tong Xiao, and Jingbo Zhu. 2020. A simple and effective approach to robust unsupervised bilingual dictionary induction. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5990–6001, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Bilingual word representations with monolingual quality in mind. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 151–159. Association for Computational Linguistics.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26*, pages 3111–3119.

Tasnim Mohiuddin, M Saiful Bari, and Shafiq Joty. 2020. LNMap: Departures from isomorphic assumption in bilingual lexicon induction through nonlinear mapping in latent space. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2712–2723, Online. Association for Computational Linguistics.

Tanmoy Mukherjee, Makoto Yamada, and Timothy Hospedales. 2018. Learning unsupervised word translations without adversaries. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 627–632, Brussels, Belgium. Association for Computational Linguistics.

Ndapa Nakashole. 2018. NORMA: Neighborhood sensitive maps for multilingual word embeddings. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 512–522, Brussels, Belgium. Association for Computational Linguistics.

Ndapa Nakashole and Raphael Flauger. 2018. Characterizing departures from linearity in word translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 221–227, Melbourne, Australia. Association for Computational Linguistics.

Aitor Ormazabal, Mikel Artetxe, Gorka Labaka, Aitor Soroa, and Eneko Agirre. 2019. Analyzing the limitations of cross-lingual word embedding mappings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4990–4995, Florence, Italy. Association for Computational Linguistics.

Barun Patra, Joel Ruben Antony Moniz, Sarthak Garg, Matthew R Gormley, and Graham Neubig. 2019. BLISS in non-isometric embedding spaces.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Anders Søgaard, Sebastian Ruder, and Ivan Vulić. 2018. On the limitations of unsupervised bilingual dictionary induction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 778–788. Association for Computational Linguistics.

Ivan Vulić and Marie-Francine Moens. 2016. Bilingual distributed word representations from document-aligned comparable data. *Journal of Artificial Intelligence Research*, 55(1):953–994.

Ivan Vulić, Sebastian Ruder, and Anders Søgaard. 2020. Are all good word vector spaces isomorphic? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3178–3192, Online. Association for Computational Linguistics.

Zirui Wang, Jiateng Xie, Ruochen Xu, Yiming Yang, Graham Neubig, and Jaime Carbonell. 2019. Cross-lingual alignment vs joint training: A comparative study and a simple unified framework. *arXiv preprint arXiv:1910.04708*.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Ruochen Xu, Yiming Yang, Naoki Otani, and Yuexin Wu. 2018. Unsupervised cross-lingual transfer of word embedding spaces. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2465–2474, Brussels, Belgium. Association for Computational Linguistics.

Pengcheng Yang, Fuli Luo, Shuangzhi Wu, Jingjing Xu, Dongdong Zhang, and Xu Sun. 2018. Learning unsupervised word mapping by maximizing mean discrepancy. *arXiv preprint arXiv:1811.00275*.

Meng Zhang, Yang Liu, Huanbo Luan, and Maosong Sun. 2017. Adversarial training for unsupervised bilingual lexicon induction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1959–1970, Vancouver, Canada. Association for Computational Linguistics.

Chunting Zhou, Xuezhe Ma, Di Wang, and Graham Neubig. 2019. Density matching for bilingual word embedding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1588–1598, Minneapolis, Minnesota. Association for Computational Linguistics.

Principled Paraphrase Generation with Parallel Corpora

Aitor Ormazabal[1], **Mikel Artetxe**[2], **Aitor Soroa**[1], **Gorka Labaka**[1], **Eneko Agirre**[1]
[1]HiTZ Center, University of the Basque Country (UPV/EHU)
[2]Meta AI
{aitor.ormazabal,a.soroa,gorka.labaka,e.agirre}@ehu.eus
artetxe@fb.com

Abstract

Round-trip Machine Translation (MT) is a popular choice for paraphrase generation, which leverages readily available parallel corpora for supervision. In this paper, we formalize the implicit similarity function induced by this approach, and show that it is susceptible to non-paraphrase pairs sharing a single ambiguous translation. Based on these insights, we design an alternative similarity metric that mitigates this issue by requiring the entire translation distribution to match, and implement a relaxation of it through the Information Bottleneck method. Our approach incorporates an adversarial term into MT training in order to learn representations that encode as much information about the reference translation as possible, while keeping as little information about the input as possible. Paraphrases can be generated by decoding back to the source from this representation, without having to generate pivot translations. In addition to being more principled and efficient than round-trip MT, our approach offers an adjustable parameter to control the fidelity-diversity trade-off, and obtains better results in our experiments.

1 Introduction

Paraphrase generation aims to generate alternative surface forms expressing the same semantic content as the original text (Madnani and Dorr, 2010), with applications in language understanding and data augmentation (Zhou and Bhat, 2021). One popular approach is to use an MT system to translate the input text into a pivot language and back (Wieting and Gimpel, 2018; Mallinson et al., 2017; Roy and Grangier, 2019). While it intuitively makes sense that translating to another language and back should keep the meaning of a sentence intact while changing its surface form, it is not clear what exactly would be considered a paraphrase by such a system.

In this work, we show that the probability of a paraphrase x_p given a source sentence x_s under a round-trip MT system can be naturally decomposed as $P(x_p|x_s) = P(x_p)S(x_p, x_s)$, where S is a symmetric similarity metric over the paraphrase space and $P(x_p)$ the probability of x_p. We argue that this similarity function is not appropriate in the general case, as it can assign a high score to sentence pairs that share an ambiguous translation despite not being paraphrases of each other. This phenomenon is illustrated in Figure 2, where x_s and x_p share a confounding translation without gender marker.

So as to address this issue, we design an alternative similarity function that requires the entire translation distribution to match, and develop a relaxation of it through the Information Bottleneck (IB) method. We implement this approach using an adversarial learning system depicted in Figure 1.

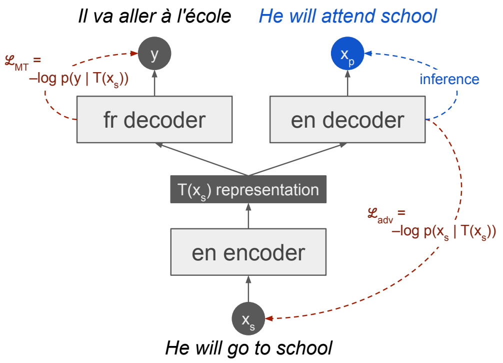


Figure 1: **Proposed system.** Given the input x_s, we aim to learn a representation $T(x_s)$ that encodes as much information as possible about it's reference translation y (ensuring that the meaning is preserved), and as little information as possible about x_s itself (ensuring that surface information is removed). We achieve this through adversarial learning, where the encoder minimizes $\lambda\mathcal{L}_{MT} - (1-\lambda)\mathcal{L}_{adv}$ and the two decoders minimize $\mathcal{L}_{MT}$ and $\mathcal{L}_{adv}$. At inference time, we couple the English encoder and decoder to generate a paraphrase x_p which, being conditioned on $T(x)$, will preserve the meaning of x_s but use a different surface form.

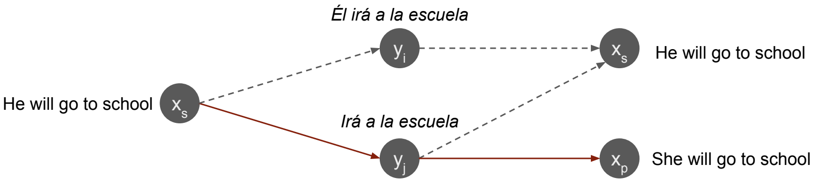


Figure 2: **Confounding translation problem in round-trip MT.** *"Irá a la escuela"* does not mark the gender of the subject due to ellipsis, and it is thus a valid translation of both *"He will go to school"* and *"She will go to school"*. As a consequence, round-trip MT could generate *"She will go to school"* as a paraphrase of *"He will go to school"*. Our approach mitigates this issue by requiring the full translation distribution to match.

Our model combines an encoder that, for a given sentence, removes the information that is not relevant to predict its translation, and a decoder that reconstructs a paraphrase from this encoding. In addition to being more principled, our approach is more efficient than round-trip MT at inference, can be tuned to favor fidelity or diversity, and achieves a better trade-off between the two. Our code is freely available [1].

2 Related Work

We next review the paraphrase generation literature (§2.1), and describe the information bottleneck method (§2.2), which is the basis of our proposal.

2.1 Paraphrase generation

Early work on paraphrasing focused on retrieval methods, either extracting plausible sentences from large corpora for generation (Barzilay and McKeown, 2001; Bannard and Callison-Burch, 2005), or identifying paraphrase pairs from weakly aligned corpora to create paraphrase datasets (Coster and Kauchak, 2011; Dolan et al., 2004). More recently, neural approaches for paraphrase generation have dominated the field. We classify these methods according to the type of supervision they use.

Monolingual corpora. These systems are trained in an unsupervised fashion using unlabeled monolingual corpora. They usually employ an information bottleneck, with the goal of encoding semantic information in the latent space. Approaches include Variational Autoencoders (VAE) (Bowman et al., 2016), VAEs with Vector Quantization (Roy and Grangier, 2019), and latent bag-of-words models (Fu et al., 2019). Huang and Chang (2021) disentangle semantic and syntactic content in the latent space through a bag of words representation, which allows for syntactically controllable generation.

Parallel corpora. These systems are trained on pairs of parallel sentences in two languages. Most of these methods are based on round-trip MT, where a sentence is translated to a pivot language and back in order to obtain a paraphrase. Hu et al. (2019) add lexical constraints to the MT decoding procedure to obtain better paraphrases. Mallinson et al. (2017) generate not one but multiple pivot sentences and use a fusion-in-decoder strategy.

Paraphrase corpora. These systems are trained in a supervised manner over pairs or clusters of paraphrases. When such data is available, training a regular sequence-to-sequence model is a strong baseline (Egonmwan and Chali, 2019). Kumar et al. (2019) add submodular optimization to improve paraphrase diversity. Some VAE-based methods also leverage paraphrase clusters to learn a latent representation that disentangles meaning and form (Iyyer et al., 2018; Kumar et al., 2020; Hosking and Lapata, 2021; Chen et al., 2019). Most of these methods require a syntactic exemplar for generation, and assume that all surface forms are valid for all sentences. Hosking and Lapata (2021) do away with this assumption in the context of question paraphrasing, predicting a valid syntactic embedding from a discrete set at test time.

While it is paraphrase corpora that offers the strongest supervision, such data is hard to obtain and usually restricted to narrow domains like Quora Question Pairs, WikiAnswers and Twitter (Hosking and Lapata, 2021; Kumar et al., 2019; Egonmwan and Chali, 2019). In contrast, parallel corpora is widely available, while offering a stronger training signal than monolingual corpora. For that reason, round-trip MT is a common choice when paraphrases are needed for downstream tasks (Xie et al.,

[1] https://github.com/aitorormazabal/paraphrasing-from-parallel

2020; Artetxe et al., 2020), as well as a common baseline in the paraphrasing literature (Hosking and Lapata, 2021; Roy and Grangier, 2019).[2] Our work focuses on this class of systems, identifying the limitations of round-trip MT and proposing a more principled alternative.

2.2 The Information Bottleneck Method

Given two random variables X, Y, the Information Bottleneck (IB) method (Tishby et al., 1999) seeks to learn a representation $T(X)$ that minimizes the Mutual Information (MI) between T and X, while preserving a minimum MI between T and Y. That is, the objective $I(X,T)\ s.t.\ I(T,Y) \geq \gamma$ is minimized. Since the MI is usually impossible to calculate exactly for neural representations, a common approach is to use variational methods, that turn the estimation problem into an optimization one. This can be done by adding a neural decoder on top of the representation, and training the entire system end-to-end (Poole et al., 2019). This is the approach we follow in this work.

3 Characterizing Round-trip MT

Let X be a random variable representing a sequence in the source language, and Y be a random variable representing its translation into a pivot language.[3] Given an input sequence $x_s \in X$, we can use round-trip MT to generate a paraphrase $x_p \in X$ by translating x_s into the pivot language and back, according to the forward and backward translation models $P(y|x_s)$ and $P(x_p|y)$. As such, we can formulate the probability of round-trip MT generating a particular paraphrase x_p by marginalizing over the set of possible pivot translations:

$$P(x_p|x_s) = \sum_{y \in Y} P(y|x_s)P(x_p|y) \qquad (1)$$

In what follows, we will characterize the paraphrases produced by this approach, i.e. the properties that x_p needs to meet in relation to x_s for $P(x_p|x_s)$ to be high.[4]

By applying Bayes' rule, we can rewrite Eq. 1 as follows:

$$P(x_p|x_s) = P(x_p) \underbrace{\sum_{y \in Y} \frac{P(y|x_s)P(y|x_p)}{P(y)}}_{S_{MT}(x_p,x_s)} \qquad (2)$$

The sum on the right hand side can be interpreted as a symmetric similarity function, $S_{MT}(x_p, x_s) = S_{MT}(x_s, x_p) = \sum_y \frac{P(y|x_s)P(y|x_p)}{P(y)}$, which measures the likelihood of two sentences to be actual paraphrases. The probability of x_p given x_s then becomes $P(x_p|x_s) = P(x_p)S_{MT}(x_p, x_s)$, which is the similarity between x_s and x_p, weighted by the marginal probability of x_p.

But when are x_s and x_p considered similar according to the above definition of $S_{MT}(x_s, x_p)$? Intuitively, S_{MT} is a measure of the *overlap* between the conditional distributions that x_s and x_p induce over Y. This will be highest when $P(y|x_s)P(y|x_p)$ is as large as possible for as many y as possible. At the same time, $P(y|x_s)P(y|x_p)$ will be high when both $P(y|x_s)$ and $P(y|x_p)$ are high, that is, when y is a probable translation of both x_s and x_p. This captures the intuition that two sentences are similar when they can be translated into the same text in the pivot language.

But what if x_s and x_p have one particular high-probability translation y_j in common, but differ in the rest? As illustrated in Figure 2, this can happen when y_j is ambiguous in the target language and can mean both x_s and x_p, even if x_s and x_p are not equivalent (e.g., when x_s uses the masculine form, x_p the feminine form, and y_j does not mark the gender). In this case, the sum $\sum_y \frac{P(y|x_s)P(y|x_p)}{P(y)}$ will be dominated by $\frac{P(y_j|x_s)P(y_j|x_p)}{P(y_j)}$, which will be high when both $P(y_j|x_s)$ and $P(y_j|x_p)$ are high.

We can thus conclude that the implicit similarity function underlying round-trip MT is flawed, as it assigns a high score to a pair of sequences (x_s, x_p) that have an ambiguous translation in common. As a consequence, round-trip MT will generate x_p as a paraphrase of x_s with a high probability, even if the two sequences have a different meaning.

4 Principled Paraphrasing

As shown in the previous section, the implicit similarity function induced by round-trip MT is not adequate in the general case, as it assigns a high score to pairs of sequences that share a single translation, despite differing in the rest. So as to address

[2] Round-trip MT has also been used to generate synthetic paraphrase corpora (Wieting and Gimpel, 2018).

[3] For convenience, we will also use X and Y to refer to the set of source and target language sequences, and abbreviate probabilities of the form $P(X = x)$ as $P(x)$.

[4] Some round-trip MT systems do not consider all possible translations into the pivot language, but only a subset of them (Mallinson et al., 2017). In that case, the sum in Eq. 1 goes over $y \in \{y_1, ..., y_K\}$, and we need to introduce a partition $Z = \sum_{y \in \{y_1,...,y_K\}} P(y|x_s)$ to normalize the probabilities. However, the fundamental analysis in this section still applies. Refer to Appendix A for more details.

this, we can define an alternative similarity function that requires the entire translation distribution to match:

$$S(x_p, x_s) = \begin{cases} 1 & P(y|x_p) = P(y|x_s) \forall y \in Y \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

and use it to replace S_{MT} in Eq. 2 so that $P(x_p|x_s) \propto P(x_p)S(x_p, x_s)$. However, this definition is too strict, as it is virtually impossible that $P(y|x_p)$ and $P(y|x_s)$ are exactly the same for all $y \in Y$.[5] In 4.1, we define a relaxation of it through the IB method, which introduces an adjustable parameter β to control how much we deviate from it. In 4.2, we characterize the paraphrases generated by this approach, showing that they are less susceptible to the problem of confounding translations described in the previous section.

4.1 IB-based relaxation

So as to implement the similarity function in Eq. 3, we will use the IB method to learn an encoding T for X such that the following holds:

$$S(x_p, x_s) = \frac{P(x_p|T(x_s))}{P(x_p)Z(x_s)} \tag{4}$$

where $Z(x_s)$ is a normalizer that does not depend on the paraphrase candidate x_p.

As seen in §2.2, given a source variable X and a target variable Y, the IB method seeks to find an encoding $T(X)$ that minimizes the MI with X (maximizing compression), while preserving a certain amount of information about Y:

$$\min_T I(X, T) \; s.t \; I(T, Y) \geq \gamma. \tag{5}$$

This constrained minimization is achieved by introducing a Lagrange multiplier β and minimizing

$$\min_T I(X, T) - \beta I(T, Y). \tag{6}$$

As $\beta \to \infty$, all the information about Y is preserved and the IB method learns a minimal sufficient statistic T, that is, an encoding that satisfies $I(T, Y) = I(X, Y)$ while achieving the lowest $I(T, X)$ possible. The following theorem states that such a minimal sufficient statistic T induces the similarity function in Eq. 3 (proof in Appendix C):

Theorem 1. *Suppose the random variable X represents a sentence in the source language, Y represents its translation, and T is a minimal sufficient statistic of X with respect to Y. Let x_p and x_s be a pair of sentences in the source language. Then, $P(x_p|T(x_s)) = P(x_p)\frac{S(x_p,x_s)}{Z(x_s)}$, where S is given by Equation 3, and Z is a normalizing factor that does not depend on x_p.*

Thus, as $\beta \to \infty$ the IB method approximates the similarity metric S. In practice, when β is set to a fixed finite number, losing some information about the target variable is allowed, and a relaxation of the metric S is learned instead.

4.2 Characterizing IB-based paraphrasing

We will next analyze the relaxation of S induced by the IB method. We will characterize what kind of sentences are considered paraphrases by it, showing that it is less susceptible to the problem of confounding translations found in round-trip MT (§3). Derivations for the results in this section, as well as alternative bounds and broader discussion can be found in Appendix B.

As seen in §4.1, we define paraphrase probabilities given an encoding T as $P(x_p|T(x_s)) = P(X = x_p|T(X) = T(x_s))$, which can only be non-zero if $T(x_p) = T(x_s)$. This means that the encoding T will partition the source space into a collection of paraphrase clusters according to its value. Mathematically, given the equivalence relation $x_1 \sim x_2 \iff T(x_1) = T(x_2)$, only sentence pairs within the same equivalence class will have non-zero paraphrase probabilities. We then have the following theorem:

Theorem 2. *Suppose T is a solution of the IB optimization problem $\min_T I(X, T) \; s.t \; I(T, Y) \geq \gamma$, and $\epsilon = I(X, Y) - \gamma$. If $\mathcal{A}$ is the partition on X induced by T, we have:*

$$\sum_{A \in \mathcal{A}} \max_{x_1, x_2 \in A} \frac{P(x_1)P(x_2)}{2(P(x_1) + P(x_2))} \cdot D_1(P_{Y|x_1}, P_{Y|x_2})^2 \leq \epsilon, \tag{7}$$

where D_1 is the L_1 norm distance.

It is easy to see that, when $\epsilon = 0$, corresponding to $\gamma = I(X, Y)$ and $\beta \to \infty$, this forces all distances to be zero. In that case, only sentences with identical translation distributions are considered paraphrases, in accordance with Theorem 1.

In the general case, Theorem 2 states that the L_1 distance between the translation distributions of

[5] One reason is that we use empirical estimates of $P(y|x_p)$ and $P(y|x_s)$, which will deviate from the ground truth.

sentences that are considered paraphrases cannot be high, as it will be bounded by a function of ϵ. While the S_{MT} metric in §3 can be dominated by a high-probability term and effectively ignore differences in probability for the less likely translations, the L_1 norm gives equal importance to differences in probability for every translation candidate. Thanks to this, the resulting system will be less susceptible to the problem of confounding translations.

5 Proposed System

In this section, we describe a practical implementation of the IB-based paraphrasing approach defined theoretically in §4.

As illustrated in Figure 1, our system can be seen as an extension of a regular encoder-decoder MT architecture with an additional adversarial decoder, which is trained with an auto-encoding objective to reconstruct the original sentence x_s from the encoder representation $T(x_s)$. The encoder is trained to minimize the cross-entropy loss of the MT decoder, while maximizing the loss of the adversarial decoder. This way, the encoder is encouraged to remove as much information about x_s as possible, while retaining the information that is necessary to predict its reference translation y. Thanks to this, $T(x_s)$ should capture the semantic content of x_s (which is relevant to predict y), without storing additional surface information (which is not relevant to predict y). Once the model is trained, the adversarial decoder can be used to generate paraphrases of x_s from this representation $T(x_s)$.

This adversarial architecture can be interpreted as an implementation of the IB method as follows. Following Poole et al. (2019), we start by adding a decoder $q(y|t)$ on top of the encoder $T(x)$, and rewrite the $I(T,Y)$ term as:

$$\begin{aligned} I(T,Y) &= \mathbb{E}_{P(y,t)}\left[\log\frac{q(y|t)}{P(y)}\right] \\ &\quad + \mathbb{E}_{P(t)}[KL(P(y|t)||q(y|t))] \\ &\geq \mathbb{E}_{P(y,t)}\Big[\log q(y|t)\Big] + h(Y), \end{aligned} \tag{8}$$

where equality will hold if q is the true conditional distribution $q(y|t) = P(y|t)$, and h is the differential entropy. If we parametrize T and q by a neural network encoder-decoder architecture the $\mathbb{E}_{P(y,t)}\Big[\log q(y|t)\Big]$ term in Eq. 8, can be rewritten as $\mathbb{E}_{P(y,x)}\Big[\log q(y|T(x))\Big]$, which is precisely the log likelihood of the data distribution of X,Y given by P. In other words, by training the encoder-decoder to maximize Eq. 8, we are implicitly maximizing the mutual information $I(T,Y)$.

Similarly, one can approximate

$$\begin{aligned} I(X,T) &\geq \mathbb{E}_{P(x,t)}\Big[\log q(x|t)\Big] + h(X) \\ &= \mathbb{E}_{P(x)}\Big[\log q(x|T(x))\Big] + h(X), \end{aligned} \tag{9}$$

where equality will hold when q is the true conditional distribution and $q(x|T(x)) = P(x|T(x))$. Thus, given an ideal decoder q that perfectly approximates the conditional distributions $q(x|T(x))$ and $q(y|T(x))$, the IB minimization problem is equivalent to minimizing

$$\begin{aligned} &\mathbb{E}_{p(x)}\Big[\log q(x|T(x))\Big] - \beta\mathbb{E}_{P(y,t)}\Big[\log q(y|t)\Big] \\ &= \mathbb{E}_{P(x,y)}\Big[\log q(x|T(x)) - \beta\log q(y|T(x))\Big]. \end{aligned} \tag{10}$$

In practice, we parametrize both the encoder T and the decoder q with transformer neural networks, and learn them from a parallel corpus. Since $\log q(y|T(x))$ is a lower bound of $I(T,Y) - h(Y)$, maximizing this term is theoretically sound. Minimizing $\mathbb{E}_{P(x)}\Big[\log q(x|T(x))\Big]$, on the other hand, amounts to minimizing a lower bound, which, while not as theoretically solid, is common practice in the variational optimization literature (Chen et al., 2018; Kim and Mnih, 2018).

Finally, we reparametrize Eq. 10 by setting $\lambda = \frac{\beta}{1+\beta}$ to obtain the equivalent minimization objective

$$\begin{aligned} \mathcal{L}(T,q) &= \mathbb{E}_{P(x,y)}[-\lambda\log q(y|T(x)) \\ &\quad +(1-\lambda)\log q(x|T(x))] = \\ &\lambda\mathcal{L}_{MT}(T,q) - (1-\lambda)\mathcal{L}_{Adv}(T), \end{aligned} \tag{11}$$

where $\mathcal{L}_{MT}$ is the regular MT loss of cross-entropy with the translation target, and $\mathcal{L}_{Adv}$ is the cross-entropy with the source sentence (see Figure 1).[6] We thus observe that the proposed adversarial training architecture approximates the IB method. The

[6] We make the adversarial term a function of T only in the minimization objective, as the gradient from the adversarial term is only propagated to the encoder. The adversarial decoder is independently trained to predict the source from the encoded representation.

setting $\beta \to \infty$ corresponds to $\lambda \to 1$, where the optimal solution is a minimal sufficient statistic.

During training, the expectation in Eq. 11 is approximated by sampling batches from the training data. Care must be taken when optimizing the loss, as we do not want to propagate gradients of the adversarial loss to the adversarial decoder. If we did, a trivial way to minimize $(1 - \lambda) \log q(x|T(x))$ would be to make the decoder bad at recovering x, which would not encourage $T(x)$ to encode as little information as possible. To prevent this, we use a percentage K of the batches to learn the adversarial decoder $\log q(x|T(x))$, where the encoder is kept frozen. The rest of the batches are used to optimize the full term $-\lambda \log q(y|T(x)) + (1 - \lambda) \log q(x|T(x))$, but the gradients for the second term are only propagated to the encoder.

6 Experimental Design

We experiment with the following systems:

- **Proposed.** Our system described in §5. We share the weights between the MT decoder and the adversarial decoder, indicating the language that should be decoded through a special language ID token. Unless otherwise indicated, we use $\lambda = 0.73$ and $K = 0.7$, which performed best in the development set.[7]
- **Round-trip MT.** A baseline that uses two separate MT models to translate into a pivot language and back (see §3).
- **Copy.** A baseline that copies the input text.

We use mBART (Liu et al., 2020) to initialize both our proposed system and round-trip MT, and train them using the same hyper-parameters as in the original work.[8] In both cases, we use the English-French WMT14 dataset (Bojar et al., 2014) as our parallel corpus for training.[9] We report results for two decoding strategies: beam search with a beam size of 5, and top-10 sampling with a temperature of 0.9 (optimized in the development set).[10]

Model	Self-BLEU ↓ (diversity)	BLEU ↑ (fidelity)	iBLEU ↑ (combined)
Copy	100.0	23.0	-13.9
MT (beam)	51.1	18.8	-2.17
MT (sampling)	41.4	15.8	-1.36
Ours (beam)	33.0	15.5	0.95
Ours (sampling)	27.3	13.2	**1.05**
Human	18.1	19.8	8.43

Table 1: **Results on the MTC dataset for three baselines (top rows), our two systems, and human performance.** ↓ smaller is better, ↑ larger is better.

We consider two axes when evaluating paraphrases: *fidelity* (the extent to which the meaning of the input text is preserved) and *diversity* (the extent to which the surface form is changed). Following common practice, we use a corpus of gold paraphrases to automatically measure these. More concretely, given the source sentence s, the reference paraphrase r and the candidate paraphrase c, we use BLEU(c, r) as a measure of fidelity, and BLEU(c, s)—known as self-BLEU—as a measure of diversity. An ideal paraphrase system would give us a high BLEU, with as low a self-BLEU as possible. Given that there is generally a tradeoff between the two, we also report iBLEU = α BLEU $-(1-\alpha)$ self-BLEU, which combines both metrics into a single score (Mallinson et al., 2017). Following Hosking and Lapata (2021), we set $\alpha = 0.7$.

For development, we extracted 156 paraphrase pairs from the STS Benchmark dataset (Cer et al., 2017), taking sentence pairs with a similarity score above 4.5. For our final evaluation, we used the Multiple Translations Chinese (MTC) corpus (Huang et al., 2002), which comprises three sources of Chinese journalistic text translated into English by multiple translation agencies. We extract the translations of the first two agencies to obtain an test set of 993 paraphrase pairs, where one is the source and the other the reference paraphrase. The third sentence if kept as an additional paraphrase for estimating human performance.

7 Results

We next report our main results (§7.1), followed by a qualitative analysis (§7.2).

7.1 Main results

We report our main results in Table 1. As it can be seen, our proposed system outperforms all baselines in terms of iBLEU, indicating that it achieves

[7] We performed a grid search, where $\lambda \in \{0.7, 0.73, 0.8\}$ and $K \in \{0.7, 0.8\}$, and chose the checkpoint with best iBLEU with $\alpha = 0.7$.

[8] 0.3 dropout, 0.2 label smoothing, 2500 warm-up steps, $3e-5$ maximum learning rate, and $100K$ total steps.

[9] We filter the dataset by removing sentence pairs with a source/target length ratio that exceeds 1.5 or are longer than 250 words.

[10] In the case of round-trip MT, we always use beam search to generate the pivot translation, and compare the two approaches to generate paraphrases from it.

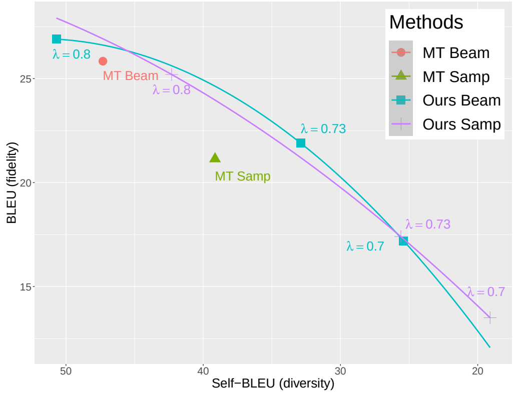


Figure 3: **Effect of varying the λ parameter on the development set.** BLEU in the vertical axis. The horizontal self-BLEU axis is mirrored, so systems toward the top right have the best trade-off between diversity and fidelity.

a better trade-off between diversity and fidelity. This advantage comes from a large improvement in diversity as measured by self-BLEU, at a cost of a small drop in fidelity as measured by BLEU. Both for round-trip MT and our proposed system, beam search does better than sampling in terms of fidelity, at the cost of sacrificing in diversity. Finally, the human reference scores show ample room for improvement in both axes.

While our proposed system achieves the best combined score, our results also show that different approaches behave differently in terms of diversity and fidelity. In practice, it would be desirable to have a knob to control the trade-off between the two, as one may want to favor diversity or fidelity depending on the application. One additional advantage of our approach over round-trip MT is that it offers an adjustable parameter λ to control the trade-off between these two axes. So as to understand the effect of this parameter, we tried different values of it in the development set, and report the resulting curve in Figure 3 together with the MT baselines. BLEU and Self-BLEU scores of the best checkpoints for each λ (0.7,0.73,0.8) and plot the results together with the MT baselines for our systems in Figure 3.

As expected, higher values of λ yield systems that tend to copy more, being more faithful but less diverse. Consistent with our test results, we find that, for a given value of λ, beam search does better than sampling in terms of fidelity, but worse in terms of diversity, yet both decoding strategies can be adjusted to achieve a similar trade-off. More importantly, we observe that both curves are above round-trip MT, the gap being largest for the sampling variant. We can thus conclude that our proposed approach does better than round-trip MT for a comparable trade-off between diversity and fidelity, while offering a knob to adjust this trade-off as desired.

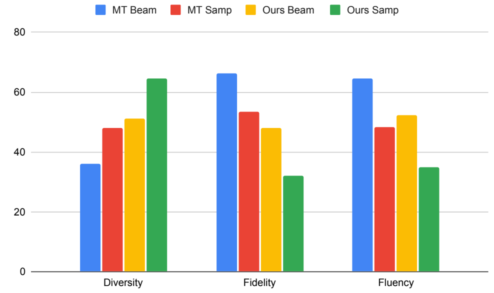


Figure 4: **Human evaluation results (larger is better)**. Refer to §7.2 for more details.

7.2 Qualitative analysis

So as to better understand the behavior of our approach in comparison with round-trip MT, we carried out a human evaluation through Amazon Mechanical Turk. Following the setup of Hosking and Lapata (2021), we sample 200 sentences from the MTC corpus and generate a pair of paraphrases for each of them, randomly choosing two systems to generate them. We then ask human evaluators to compare the two sentences according to three criteria: diversity, fidelity and fluency. More details about the judging criteria can be found in Appendix D.

Figure 4 reports the percentage of head-to-head comparisons that each system has won. The results that we obtain are consistent with the trends observed in §7.1. More concretely, we observe that the beam search variant of round-trip MT achieves the best results in terms of fluency and fidelity, but does worst in diversity, indicating a tendency to copy. Our method with beam search does slightly better than the sampling MT variant in terms of diversity and slightly worse in terms of fidelity —indicating a tendency to favor diversity over fidelity— while also being more fluent. Finally, the sampling variant of our method achieves the best diversity, but has the worst fidelity and fluency.

So as to further contrast these results, we manu-

Original	MT (beam)	MT (sampling)	Ours (beam)	Ours (sampling)
The index would fall 3.9% if the sales of vehicles were not included.	The index would fall by 3.9 per cent if vehicle sales were not included.	The Index would fall 3.9% if the vehicle sales were not included, or 3.9% if the vehicle sales were excluded.	The index would decline by 3.9% if the vehicle sales were not included.	The index would decline by 3.9% if vehicles were not included in sales.
Some people worry that this will affect the business of large-sized Canadian enterprises.	There are concerns that this may affect the operations of large Canadian companies.	There is concern that this may affect what large Canadian firms have in their business.	Some may be concerned that this will affect the major Canadian enterprises.	Some people worry that this will impact on the great enterprises in Canada.
These people can set examples and they can have direct influence over the improvement of local human rights conditions and the protection of employees.	These individuals can provide examples and have a direct influence on the improvement of local human rights and employee protection conditions.	These may lead to examples and direct influence on the betterment of local human rights conditions and on the protection of wage earners.	They can provide examples and can directly influence the improvement of local human rights conditions and the protection of employees.	They can provide examples and have direct influence on improving local human rights and the protection of employees' conditions.
The National Youth League said these activities are aimed at showing support and adoration for the state leaders.	The National Youth League stated that these activities are aimed at showing support and admiration to State leaders.	The National Youth League (NDY) stated that these activities are aimed at demonstrating support and admiration for State leaders.	The National League of Youth indicated that these activities are intended to provide support and encourage state leaders.	The National Youth League has stated that such activities are aimed at showing support and admiration for State leaders.

Table 2: Sample paraphrases generated by the different methods.

ally analyzed some paraphrases,[11] and report some examples in Table 2. Just in line with our previous results, we observe that the beam search variant of round-trip MT tends to deviate the least from the original sentence, while the sampling variant of our method generates the most diverse paraphrases (e.g., changing *"sales of vehicles were not included"* to *"vehicles were not included in sales"*). At the same time, we observe that this tendency to improve diversity can cause artifacts like paraphrasing named entities (e.g., changing *"National Youth League"* to *"National League of Youth"*), which can partly explain the drop in fidelity.

8 Conclusions

In this work, we have shown that the implicit similarity function present in round-trip MT is not appropriate in the general case, as it considers sentence pairs that share a single ambiguous translation to be paraphrases. We address this issue by designing an alternative similarity function that requires the entire translation distribution to match, and develop a relaxation of it through the IB method, which we prove to be less susceptible to the problem of confounding translations. We implement this approach through adversarial learning, training an encoder to preserve as much information as possible about the reference translation, while encoding as little as possible about the source. Not only is our approach more principled than round-trip MT, but it is also more efficient at inference, as it does not need to generate an intermediate translation. In addition, it offers a knob to adjust the fidelity-diversity trade-off through the λ parameter, and obtains strong results in our experiments, outperforming round-trip MT.

Acknowledgments

Aitor Ormazabal, Gorka Labaka, Aitor Soroa and Eneko Agirre were supported by the Basque Government (excellence research group IT1343-19 and DeepText project KK-2020/00088) and the Spanish MINECO (project DOMINO PGC2018-102041-B-I00 MCIU/AEI/FEDER, UE). Aitor Ormazabal was supported by a doctoral grant from the Spanish MECD. Computing infrastructure funded by UPV/EHU and Gipuzkoako Foru Aldundia.

References

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2020. Translation artifacts in cross-lingual transfer learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*,

[11] We randomly sampled 20 sentences from MTC and chose four illustrative examples for Table 2. The 20 random examples are shown in Appendix E.

pages 7674–7684, Online. Association for Computational Linguistics.

Colin Bannard and Chris Callison-Burch. 2005. Paraphrasing with bilingual parallel corpora. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, page 597–604, USA. Association for Computational Linguistics.

Regina Barzilay and Kathleen R. McKeown. 2001. Extracting paraphrases from a parallel corpus. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, ACL '01, page 50–57, USA. Association for Computational Linguistics.

Ondřej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, et al. 2014. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the ninth workshop on statistical machine translation*, pages 12–58.

Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew Dai, Rafal Jozefowicz, and Samy Bengio. 2016. Generating sentences from a continuous space. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 10–21, Berlin, Germany. Association for Computational Linguistics.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada. Association for Computational Linguistics.

Mingda Chen, Qingming Tang, Sam Wiseman, and Kevin Gimpel. 2019. Controllable paraphrase generation with a syntactic exemplar. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5972–5984, Florence, Italy. Association for Computational Linguistics.

Ricky T. Q. Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. 2018. Isolating sources of disentanglement in variational autoencoders. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

William Coster and David Kauchak. 2011. Simple English Wikipedia: A new text simplification task. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 665–669, Portland, Oregon, USA. Association for Computational Linguistics.

Bill Dolan, Chris Quirk, and Chris Brockett. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 350–356, Geneva, Switzerland. COLING.

Elozino Egonmwan and Yllias Chali. 2019. Transformer and seq2seq model for paraphrase generation. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 249–255, Hong Kong. Association for Computational Linguistics.

Yao Fu, Yansong Feng, and John P Cunningham. 2019. Paraphrase generation with latent bag of words. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Tom Hosking and Mirella Lapata. 2021. Factorising meaning and form for intent-preserving paraphrasing.

J. Edward Hu, Rachel Rudinger, Matt Post, and Benjamin Van Durme. 2019. Parabank: Monolingual bitext generation and sentential paraphrasing via lexically-constrained neural machine translation. *CoRR*, abs/1901.03644.

Kuan-Hao Huang and Kai-Wei Chang. 2021. Generating syntactically controlled paraphrases without using annotated parallel pairs. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1022–1033, Online. Association for Computational Linguistics.

Shudong Huang, David Graff, and George R. Doddington. 2002. Multiple-translation chinese corpus.

Mohit Iyyer, John Wieting, Kevin Gimpel, and Luke Zettlemoyer. 2018. Adversarial example generation with syntactically controlled paraphrase networks. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1875–1885, New Orleans, Louisiana. Association for Computational Linguistics.

Hyunjik Kim and Andriy Mnih. 2018. Disentangling by factorising. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2649–2658. PMLR.

Ashutosh Kumar, Kabir Ahuja, Raghuram Vadapalli, and Partha Talukdar. 2020. Syntax-Guided Controlled Generation of Paraphrases. *Transactions of the Association for Computational Linguistics*, 8:330-345.

Ashutosh Kumar, Satwik Bhattamishra, Manik Bhandari, and Partha Talukdar. 2019. Submodular optimization-based diverse paraphrasing and its effectiveness in data augmentation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3609–3619, Minneapolis, Minnesota. Association for Computational Linguistics.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation.

Nitin Madnani and Bonnie J Dorr. 2010. Generating phrasal and sentential paraphrases: A survey of data-driven methods. *Computational Linguistics*, 36(3):341–387.

Jonathan Mallinson, Rico Sennrich, and Mirella Lapata. 2017. Paraphrasing revisited with neural machine translation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 881–893, Valencia, Spain. Association for Computational Linguistics.

Ben Poole, Sherjil Ozair, Aaron Van Den Oord, Alex Alemi, and George Tucker. 2019. On variational bounds of mutual information. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5171–5180. PMLR.

Aurko Roy and David Grangier. 2019. Unsupervised paraphrasing without translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6033–6039, Florence, Italy. Association for Computational Linguistics.

Naftali Tishby, Fernando C. Pereira, and William Bialek. 1999. The information bottleneck method. pages 368–377.

John Wieting and Kevin Gimpel. 2018. ParaNMT-50M: Pushing the limits of paraphrastic sentence embeddings with millions of machine translations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 451–462, Melbourne, Australia. Association for Computational Linguistics.

Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. 2020. Unsupervised data augmentation for consistency training. In *Advances in Neural Information Processing Systems*, volume 33, pages 6256–6268. Curran Associates, Inc.

Jianing Zhou and Suma Bhat. 2021. Paraphrase generation: A survey of the state of the art. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5075–5086, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

A Round-trip MT with restricted sampling

Our formulation in Section 3 considers all possible translations into the pivot language. In practice, some round-trip MT systems use restricted sampling, considering only a subset of Y. For example, ParaNET (Mallinson et al., 2017) takes the K highest probability translations given by beam search. As we show next, the fundamental analysis in Section 3 still holds in that case, and the problem of confounding translations can even be exacerbated by it.

More concretely, using this pivot selection strategy yields the following adjusted paraphrase probability:

$$P(x_p|x_s) = P(x_p) \sum_{y\in\{y_1,...,y_K\}} \frac{P(y|x_s)P(y|x_p)}{ZP(y)}, \tag{12}$$

where $\{y_1, ..., y_K\}$ are the top translation candidates and $Z = \sum_{y\in\{y_1,...,y_K\}} P(y|x_s)$. In general, if a subset $S(x_s) \in Y$ of the translation space is considered as pivots, the paraphrase probability will be

$$P(x_p|x_s) = P(x_p) \sum_{y\in S(x_s)} \frac{P(y|x_s)P(y|x_p)}{Z(x_s)P(y)} = \frac{P(x_p)}{Z(x_s)} S'_{MT}(x_p, x_s), \tag{13}$$

where $Z(x_s) = \sum_{y\in S(x_s)} P(y|x_s)$ is a normalizing factor that doesn't depend on the paraphrase x_p, and $S'_{MT}(x_p, x_s)$, is the same similarity function as S_{MT}, with the sum over y restricted to $S(x_s)$.

Using restricted pivot selection strategies such as beam search, top-K sampling, or nucleus sampling (Holtzman et al., 2020) will yield different pivot subsets $S(x_p)$, which will lead to paraphrase probabilities being assigned based on a limited subset of the entire translation distribution. This can exacerbate the issues outlined in Section 3, where the similarity metric can be dominated by a single shared high-probability translation. For example, in the case of translating from a gendered to a genderless language, while the highest probability translation will be genderless, a lower probability candidate might identify the gender, so skipping this translation sampling would increase the similarity of sentences that differ only in gender.

B Characterizing the encoding learned through the IB method

Since we will not learn a perfect minimal sufficient statistic in practice, it is desirable to characterize what the relaxation of S implemented by the IB method can learn.

To that end, we will characterize the kind of encoding T that is allowed by a given γ. Since T is a function of X, we know that $I(X,Y) \geq I(T,Y)$, and thus the condition $I(T,Y) \geq \gamma$ can be rewritten as $I(X,Y) - I(T,Y) \leq \epsilon$, where $\epsilon \geq 0$, which is the form we use throughout this section.

Now, for matters of conditional translation probabilities $P(y|T(x))$, the encoding T can be fully characterized by the equivalence relation it defines on X, where $x_1 \sim x_2$ iff $T(x_1) = T(x_2)$. Two sentences will induce the same conditional translation distribution $P(y|T(x))$ when they are clustered into the same equivalence class by T.

Theorem 3. *Let $\mathcal{S}$ denote the partition of X induced by the encoding T. We denote the elements of a cluster $S \in \mathcal{S}$ by $S = \{x_1^S, ..., x_{m_S}^S\}$. Then, the information loss $I(X,Y) - I(T,Y)$ is given by:*

$$I(X,Y) - I(T,Y) = \sum_{S\in\mathcal{S}} P(x_1^S) KL(P_{Y|x_1^S} || P_{Y|S}) \\ +... + P(x_{m_S}^S) KL(P_{Y|x_{m_S}^S} || P_{Y|S}), \tag{14}$$

and the translation probabilities conditioned on a cluster are given by the mixture distribution $P(y|T(x)) = P(y|x \in S) = \alpha_1 P(y|x_1^S) + ... + \alpha_{m^S} P(y|x_{m^S}^S)$, where $\alpha_i = \frac{P(x_i^S)}{P(x_1^S)+...+P(x_{m^S}^S)}$.

The proof can be found in Section C.2. This theorem expresses the information loss of an encoding T, $I(X,Y) - I(T,Y)$, in terms of the KL divergences between the translation distributions of source sentences $P(y|x)$ and the translation distributions given their encodings $P(y|T(x)) = P(y|x \in S)$. Intuitively, if T clusters together two sentences x_1 and x_2 (i.e. $T(x_1) = T(x_2)$ holds), such that $P_{Y|x_1}$ and $P_{Y|x_2}$ are very different, then the mixture distribution $P_{Y|S}$ will be different from both, and thus the information loss will be large.

We will now obtain more intuitive bounds for the information loss. As seen before, the translation distribution given a cluster $P(y|x \in S)$ can be expressed as a mixture of the individual translation distributions for sentences in the cluster:

$$P(y|x \in S) = \frac{P(x_1^S)}{P(x_1^S) + ... + P(x_{m^S}^S)} P(y|x_1^S) \\ +... + \frac{P(x_{m^S}^S)}{P(x_1^S) + ... + P(x_{m^S}^S)} P(y|x_{m_S}^S). \tag{15}$$

We can also define mixtures of all the distributions $P(y|x_i^S)$ except one, with the same weights as in $P(y|x \in S)$, except for a re-normalization constant. Explicitly, we define:

$$P_j^S(y) = \\ \sum_{i=1, i\neq j}^{m_S} \frac{P(x_i^S) P(y|x_i^S)}{P(x_1^S) + ... + \widehat{P(x_j^S)} + ... + P(x_{m^S}^S)}, \tag{16}$$

where the hat indicates that that element is skipped. Then, we have the following theorem:

Theorem 4. *Let $\mathcal{S}$ be the partition imposed by the encoding function T on X. We denote the elements of a cluster $S \in \mathcal{S}$ by $S = \{x_1^S, ..., x_{m_S}^S\}$. We define the partial mixtures $P_j^S(y)$ as above. Then, if the information loss satisfies $I(X,Y) - I(T,Y) \leq \epsilon$, we have*

$$\sum_{S\in\mathcal{S}} \sum_{i=1}^{m^S} \frac{P(x_i^S)(\beta_i^S)^2}{2} D_1(P_{Y|x_i^S}, P_i^S)^2 \leq \epsilon, \tag{17}$$

where $\beta_j^S = \frac{P(x_1^S)+...+\widehat{P(x_j^S)}+...+P(x_{m^S}^S)}{P(x_1^S)+...+P(x_{m^S}^S)}$ and D_1 is the L^1 norm distance.

The proof can be found in Section C.3. Intuitively, this states that, if the encoding T clusters a set of sentences $x_1, ..., x_n \in S$ together, then the translation distribution for an element $x_i \in S$, $P_{Y|x_i}$, cannot be too far from the mixture of the rest of the distributions $P_{Y|x_j}$, with $j \neq i$.

In the scenario where there are only two sentences in a cluster, $m^S = 2$, we have $P_1 = P_{Y|x_2^S}$ and $P_2 = P_{Y|x_1^S}$, and the inner sum reduces as follows (the derivation is shown in Section C.4):

$$\sum_{i=1}^{m^S} \frac{P(x_i^S)(\beta_i^S)^2}{2} D_1(P_{Y|x_i^S}, P_i^S)^2 \\ = \frac{P(x_1^S)P(x_2^S)}{2(P(x_1^S) + P(x_2^S))} D_1(P_{Y|x_1^S}, P_{Y|x_2^S})^2 \tag{18}$$

Since clustering all the elements of a set S leads to a bigger information loss than only clustering any two elements $x_1, x_2 \in S$, combining Equation 18 and Theorem 4 we obtain Theorem 2 from §4.2 as a corollary:

Theorem 2. *Suppose T is a solution of the IB optimization problem $\min_T I(X,T)$ s.t $I(T,Y) \geq \gamma$, and $\epsilon = I(X,Y) - \gamma$. If $\mathcal{S}$ is the partition on X induced by T, we have:*

$$\sum_{S \in \mathcal{S}} \max_{x_1, x_2 \in S} \frac{P(x_1)P(x_2)}{2(P(x_1)+P(x_2))} D_1(P_{Y|x_1}, P_{Y|x_2})^2 \leq \epsilon, \tag{19}$$

where D_1 is the L_1 norm distance.

This bound is the easiest to interpret intuitively, as it bounds the pairwise distances between the translation distributions of any two sentences that are considered paraphrases by the encoding T. To sum up the results from this section, the information loss allowance when learning with the IB method bounds the L^1 norm distance between the translation distributions of paraphrases. Thus the entire translation distribution is considered when learning paraphrases, potentially alleviating the problems discussed in Section 3

C Proofs

C.1 Proof of Theorem 1

We know that T is a minimal sufficient statistic of Y if and only if the following condition is satisfied:

$$\frac{P(x|y)}{P(x'|y)} \text{ independent of y} \iff T(x) = T(x') \; \forall x, x' \in X \tag{20}$$

Rewriting $P(x|y) = \frac{P(y|x)P(x)}{P(y)}$ and cancelling terms, the LHS becomes:

$$\frac{P_y(x)}{P_y(x')} = \frac{P(y|x)}{P(y|x')}\frac{P(x)}{P(x')}. \tag{21}$$

Since $\frac{P(x)}{P(x')}$ does not depend on y, the entire term will not depend on y if and only if $\frac{P(y|x)}{P(y|x')}$ is independent of y. It is easy to see that the ratio of two distributions of y will be independent of y if and only if they are the exact same distribution, and thus we can conclude that if T is a minimal sufficient statistic of Y then $T(x) = T(x') \iff P(y|x) = P(y|x') \forall y \in Y$, or, equivalently, $T(x) = T(x') \iff S(x, x') = 1$.

Thus, we have

$$\begin{aligned} P(x_p|T(x_s) &= P(X = x_p|T(X) = T(x_s)) \\ &= P(X = x_p|S(X, x_s) = 1) \\ &= \frac{P(X = x_p, S(X, x_s) = 1)}{P(S(X, x_s) = 1)} \\ &= \frac{P(X = x_p)P(S(X, x_s) = 1)|X = x_p)}{P(S(X, x_s) = 1)} \\ &= \frac{P(x_p)S(x_p, x_s)}{Z}, \end{aligned} \tag{22}$$

where $Z = P(S(X, x_s) = 1)$ is the normalizer that does not depend on x_p, as we wanted to prove. □

C.2 Proof of Theorem 3

We first expand the information loss:

$$\begin{aligned} I(X,Y) - I(T,Y) \\ &= \mathbb{E}_X KL(P_{Y|X}||P_Y) \\ &- \mathbb{E}_T KL(P_{Y|T}||P_Y) = \sum_x P(x) \\ &\Big[\sum_y P(y|x)log(P(y|x)) \\ &- P(y|x)log(P(y))\Big] \\ &- \sum_x P(x)\Big[\sum_y P(y|T(x))log(P(y|T(x)) \\ &- P(y|T(x))log(P(y))\Big] \end{aligned} \tag{23}$$

Now, for matters of conditional translation probabilities $P(y|T(x))$, the encoding T can be fully characterized by the equivalence class it defines on X, where $x_1 \sim x_2$ iff $T(x_1) = T(x_2)$. We let $\mathcal{S}$ denote the partition on X induced by this equivalence class. Then, we can rewrite:

$$\begin{aligned}
&I(X,Y) - I(T,Y)\\
&= \sum_x P(x)\Big[\sum_y P(y|x)log(P(y|x))\\
&- \ P(y|x)log(P(y))\Big]\\
&- \sum_x P(x)\Big[\sum_y P(y|T(x))log(P(y|T(x)))\\
&- P(y|T(x))log(P(y))\Big]\\
&= \sum_{S\in\mathcal{S}}\sum_{x\in S} P(x)\Big[\sum_y P(y|x)log(P(y|x))\\
&- P(y|x)log(P(y))\Big]\\
&- \sum_{S\in\mathcal{S}}\sum_{x\in S} P(x)\Big[\sum_y P(y|x\in S)log(P(y|x\in S))\\
&- P(y|x\in S)log(P(y))\Big]
\end{aligned} \tag{24}$$

For a certain $S \in \mathcal{S}$, we denote its elements by $S = \{x_1^S, ..., x_{m_S}^S\}$. Then, we have

$$\begin{aligned}
P(y|x\in S) &= \frac{P(y, x\in S)}{P(x\in S)}\\
&= \frac{P(y, x_1^S) + ... + P(y, x_{m_S}^S)}{P(x_1^S) + ... + P(x_{m_S}^S)}\\
&= \alpha_1^S P(y|x_1^S) + ... + \alpha_{m_S}^S P(y|x_{m_S}^S),
\end{aligned} \tag{25}$$

where $\alpha_i^S = \frac{P(x_i^S)}{P(x_1^S)+...+P(x_{m_S}^S)}$. We also define $\beta^S = P(x_1^S) + ... + P(x_{m_S}^S)$. Now, we can rewrite the first expression of the RHS in Equation 24:

$$\begin{aligned}
&\sum_{S\in\mathcal{S}}\sum_{x\in S} P(x)\sum_y \big[P(y|x)log(P(y|x))\\
&- P(y|x)log(P(y))\big]\\
&= \sum_{S\in\mathcal{S}}\Big[P(x_1^S)\sum_y P(y|x_1^S)log(P(y|x_1^S)) + ...\\
&+ P(x_{m_S}^S)\sum_y P(y|x_{m_S}^S)log(P(y|x_{m_S}^S))\Big]\\
&- \Big[\sum_y P(x_1^S)P(y|x_1^S)log(P(y)) + ...\\
&+ P(x_{m_S}^S)P(y|x_{m_S}^S)log(P(y))\Big]\\
&= \sum_{S\in\mathcal{S}}\Big[P(x_1^S)\sum_y P(y|x_1^S)log(P(y|x_1^S)) + ...\\
&+ P(x_{m_S}^S)\sum_y P(y|x_{m_S}^S)log(P(y|x_{m_S}^S))\Big]\\
&- \Big[\sum_y \beta^S\alpha_1^S P(y|x_1^S)log(P(y)) + ...\\
&+ \beta^S\alpha_{m_S}^S P(y|x_{m_S}^S)log(P(y))\Big]\\
&= \sum_{S\in\mathcal{S}}\Big[P(x_1^S)\sum_y P(y|x_1^S)log(P(y|x_1^S)) + ...\\
&+ P(x_{m_S}^S)\sum_y P(y|x_{m_S}^S)log(P(y|x_{m_S}^S))\Big]\\
&- \Big[\beta^S\sum_y P(y|x\in S)log(P(y))\Big]
\end{aligned} \tag{26}$$

And now we rewrite the second term in the RHS of Equation 24:

$$\begin{aligned}
&\sum_{S\in\mathcal{S}}\sum_{x\in S}P(x)\sum_{y}\big[P(y|x\in S)log(P(y|x\in S))\\
&-P(y|x\in S)log(P(y))\big]\\
&=\sum_{S\in\mathcal{S}}\Bigg[P(x_1^S)\sum_{y}P(y|x\in S)log(P(y|x\in S))\\
&+P(x_{m_S}^S)\sum_{y}P(y|x\in S)log(P(y|x\in S))\Bigg]\\
&-\Bigg[P(x_1^S)\sum_{y}P(y|x\in S)log(P(y))\\
&+P(x_{m_S}^S)\sum_{y}P(y|x\in S)log(P(y))\Bigg]\\
&=\sum_{S\in\mathcal{S}}\Bigg[\beta^S\sum_{y}P(y|x\in S)log(P(y|x\in S))\Bigg]\\
&-\Bigg[\beta^S\sum_{y}P(y|x\in S)log(P(y))\Bigg]\\
&=\sum_{S\in\mathcal{S}}\Bigg[P(x_1^S)\sum_{y}P(y|x_1^S)log(P(y|x\in S))\\
&+P(x_{m_S}^S)\sum_{y}P(y|x_{m_S}^S)log(P(y|x\in S))\Bigg]\\
&-\Bigg[\beta^S\sum_{y}P(y|x\in S)log(P(y))\Bigg],
\end{aligned}\tag{27}$$

where we have used Equation 25 in the last equality.

Substituting (26) and (27) into (24), we get:

$$\begin{aligned}
&I(X,Y)-I(T,Y)\\
&=\sum_{S\in\mathcal{S}}\Bigg[P(x_1^S)\sum_{y}P(y|x_1^S)log(P(y|x_1^S))+...\\
&+P(x_{m_S}^S)\sum_{y}P(y|x_{m_S}^S)log(P(y|x_{m_S}^S))\Bigg]\\
&-\sum_{S\in\mathcal{S}}\Bigg[P(x_1^S)\sum_{y}P(y|x_1^S)log(P(y|x\in S))\\
&+P(x_{m_S}^S)\sum_{y}P(y|x_{m_S}^S)log(P(y|x\in S))\Bigg]\\
&=\sum_{S\in\mathcal{S}}P(x_1^S)\sum_{y}\Bigg[P(y|x_1^S)log(P(y|x_1^S))\\
&-P(y|x_1^S)log(P(y|x\in S))\Bigg]+...\\
&+P(x_{m_S}^S)\sum_{y}\Bigg[P(y|x_{m_S}^S)log(P(y|x_{m_S}))\\
&-P(y|x_{m_S}^S)log(P(y|x\in S))\Bigg]\\
&=\sum_{S\in\mathcal{S}}P(x_1^S)KL(P_{Y|x_1^S}||P_{Y|S})+...\\
&+P(x_{m_S}^S)KL(P_{Y|x_{m_S}^S}||P_{Y|S})
\end{aligned}\tag{28}$$

As we wanted to show. □

C.3 Proof of Theorem 4

By Theorem 3, it is enough to show that

$$\begin{aligned}
&\sum_{S\in\mathcal{S}}P(x_1^S)KL(P_{Y|x_1^S}||P_{Y|S})\\
&\qquad+...+P(x_{m_S}^S)KL(P_{Y|x_{m_S}^S}||P_{Y|S})\\
&\qquad\geq\sum_{S\in\mathcal{S}}\sum_{i=1}^{m^S}\frac{P(x_i^S)(\beta_i^S)^2}{2}D_1(P_{Y|x_i^S},P_i^S)^2
\end{aligned}\tag{29}$$

For that, it is enough to show that

$$\begin{aligned}
&P(x_i^S)KL(P_{Y|x_i^S}||P_{Y|S})\\
&\qquad\geq\frac{P(x_i^S)(\beta_i^S)^2}{2}D_1(P_{Y|x_i^S},P_i^S)^2
\end{aligned}\tag{30}$$

for every i. Now, by Pinsker's inequality, for a given i, we have that

$$\begin{aligned}
&P(x_i^S)KL(P_{Y|x_i^S}||P_{Y|S}) = P(x_i^S) \\
&\quad KL(P_{Y|x_i^S}||\alpha_1 P_{Y|x_1^S} + ... + \alpha_{m^S} P_{Y|x_{m^S}^S}) \\
&\quad \geq \frac{P(x_i^S)}{2} D_1(\alpha_1 P_{Y|x_1^S} + ... \\
&\quad + \alpha_{m^S} P_{Y|x_{m^S}^S}, P_{Y|x_i^S})^2 \\
&\quad = \frac{P(x_i^S)}{2(P(x_1^S) + ... + P(x_{m^S}^S))^2} ||_1(P(x_1)P_{Y|x_1^S} + ... \\
&\quad + P(x_{m^S})P_{Y|x_{m^S}^S} \\
&\quad - (P(x_1^S) + ... + P(x_{m^S}^S))P_{Y|x_i^S}||^2 \\
&\quad = \frac{P(x_i^S)}{2(P(x_1^S) + ... + P(x_{m^S}^S))^2} \\
&\quad ||_1(P(x_1^S)P_{Y|x_1^S} + ...P(\widehat{x_i^S)P}_{Y|x_i^S} + ... \\
&\quad + P(x_{m^S}^S)P_{Y|x_{m^S}^S} \\
&\quad - (P(x_1^S) + ... + \widehat{P(x_i^S)} + ... + P(x_{m^S}))P_{Y|x_i^S}||^2 \\
&\quad = \frac{P(x_i^S)(P(x_1^S) + ... + \widehat{P(x_i^S)} + ... + P(x_{m^S}))^2}{2(P(x_1^S) + ... + P(x_{m^S}^S))^2} \\
&\quad ||_1 P_i^S - P_{Y|x_i^S}||^2 \\
&\quad = \frac{P(x_i^S)(\beta_i^S)^2}{2} D_1(P_i^S(Y), P_{Y|x_i^S})^2
\end{aligned} \tag{31}$$

where the hat represents that element of the sum being skipped, as we wanted to show. □

C.4 Derivation of Equation 18

$$\begin{aligned}
&\sum_{i=1}^{m^S} \frac{P(x_i^S)(\beta_i^S)^2}{2} D_1(P_{Y|x_i^S}, P_i^S)^2 \\
&= \left[\frac{P(x_1^S)(\beta_1^S)^2}{2} + \frac{P(x_2^S)(\beta_2^S)^2}{2}\right] D_1(P_{Y|x_1^S}, P_{Y|x_2^S})^2 \\
&= \left[\frac{P(x_1^S)P(x_2^S)^2}{2(P(x_1^S) + P(x_2^S))^2} + \frac{P(x_2^S)P(x_1^S)^2}{2(P(x_1^S) + P(x_2^S))^2}\right] \\
&D_1(P_{Y|x_1^S}, P_{Y|x_2^S})^2 \\
&= \left[\frac{P(x_1^S)P(x_2^S)(P(x_1^S) + P(x_2^S))}{2(P(x_1^S) + P(x_2^S))^2}\right] \\
&D_1(P_{Y|x_1^S}, P_{Y|x_2^S})^2 \\
&= \frac{P(x_1^S)P(x_2^S)}{2(P(x_1^S) + P(x_2^S))} D_1(P_{Y|x_1^S}, P_{Y|x_2^S})^2
\end{aligned} \tag{32}$$

D Human evaluation judging criteria

We ask human evaluators to compare systems on three different dimensions, according to the following instructions:

Meaning. Which of the paraphrases better preserves the meaning of the original, without adding or losing information?

Surface similarity. Which of the paraphrases is more similar compared to the original, using more similar phrasing or words? You should chose the text using more similar words or phrasing, regardless of meaning.

Fluency. Which text is a more fluent English sentence? You should choose the sentence that contains the least grammatical mistakes, and sounds more natural.

E Full paraphrase sample

We present the full list of 20 sampled paraphrases in Table 3.

Original	MT (beam)	MT (sampling)	Ours (beam)	Ours (sampling)
As of August 30th, the city had allocated a labor force of 158,300 per day in the project, with the aggregate labor contribution amounting to 936,500, and had completed 730,000 cubic meters of earth and stone.	As at 30 August, the city had allocated 158,300 persons per day to the project, with an overall contribution of 936,500 persons, and had completed the construction of 730,000 cubic metres of land and stone.	It had assigned 158,300 employees per day to the project as of 30 August, with a total staff contribution of 936,500, and it had concluded the construction of 730,000 cubic metres of earth and rock.	As of 30 August, the city had employed 158,300 persons per day for the project, a total of 936,500 of whom had been allocated for the construction of 230,000 square meters of earth and stone.	On 30 August, the City had a staff of 158,300 people per day in the project, a total of 936,500 of whom had been provided for the construction of 330,000 square foot of earth andstone work.
The United States and North Korea are scheduled to hold talks on Friday regarding US access to a suspected underground nuclear site in the vicinity of Pyongyang.	The United States and North Korea are expected to hold talks on Friday on United States access to a suspected underground nuclear site near Pyongyang.	The United States and North Korea are scheduled for talks Friday on the US access to a suspected underground nuclear site near Pyongyang.	The United States and North Korea are scheduled to hold talks Friday on the access of the United States to a suspected nearby Pyongyang ground nuclear site.	The United States and North Korea are scheduled to hold talks on Friday on United States access to a suspected nearby Pyongyang nuclear site.
For long, Xi’ning has made insufficient investment in the construction of urban infrastructure facilities, with the total investment made being only about RMB400 million during the 46-year period between the founding of the New China to 1995. As a result, the backward and underdeveloped infrastructure facilities have restricted the city’s economic development.	Xi’ning has for a long time not invested sufficiently in the construction of urban infrastructure, with a total investment of only about RMB 400 million over the 46-year period between the founding of New China and 1995, as a result of which lagging and underdeveloped infrastructure has limited the city’s economic development.	Xi’ning had for a long time not invested enough in construction of urban infrastructure, with total investment amounting only to around RMB 400m during the 46-year period between the founding of New China and 1995. Therefore, backward and underdeveloped infrastructure had impeded economic development of the city.	Since long Xi’ning’s investment in urban infrastructure has been insufficient, with a total investment of only 400 million cubic metres in total, for 46 years between the founding of the New China in 1995, thus limiting the development of infrastructure in the underdeveloped and underdeveloped areas.	Since long Xi’ning has not made adequate investment in urban infrastructure, with an overall investment of only $400 million MB over 46 years from the creation of the new China to 1995, thus restricting the development of urban infrastructure, both underdeveloped and underdeveloped.
Japan, Australia, New Zealand and South Korea Expresses Support, saying that the U.S. Has No Other Choice	Japan, Australia, New Zealand and South Korea expressed their support, stating that the United States had no other choice.	Japan, Australia, New Zealand and South Korea show their support, affirming that the United States does not have no Alternative	Japan, Australia, New Zealand and South Korea expressed their support, stating that the United States had no other choice.	Japan, Australia, New Zealand and South Korea expressed support, stating that the United States had no other choice.
Chernomyrdin also pointed out that there were also many problems in Russia last year, such as the poor performance of taxation, investment plans yet to be completed, reduced surplus in foreign trade, and ineffective fiscal and financial measures taken by the government.	Chernomyrdin also pointed out that there were also many problems in Russia last year, such as poor fiscal performance, investment plans that had not yet been completed, the reduction of the foreign trade surplus and ineffective fiscal and financial measures taken by the government.	Chernomyrdin also stressed that, in Russia, last year, there were many problems too. These included a deteriorating fiscal performance, investment plans that have yet to be completed, the reduction of foreign trade surpluses, and inefficient fiscal and financial measures taken by the Government, etc.	Chernomyrdin also noted that there were many problems in Russia last year, such as poor taxation performance, plans for investment still to be completed, the reduction in foreign trade surplus, and inadequate fiscal and financial measures taken by the Government.	Chernomyrdin further noted that there were many further problems in Russia last year, such as weak taxation performance, plans for further investment still to be completed, the reduction in foreign trade surplus, and ineffective fiscal and financial measures taken by the Government.
Based on the plan, the GDP in Russia next year is to increase by 2%, and the inflation rate is to go down to 5% to 8%.	According to the plan, Russia’s GDP will rise by 2 per cent next year and the inflation rate will rise from 5 per cent to 8 per cent.	According to the plan, Russian GDP will rise 2 per cent next year and the rate of inflation will be reduced from 5 per cent to 8 per cent.	According to this plan, GDP in Russia will increase by 2% next year and the inflation rate will drop from 5% to 8%.	Under the plan, the GDP in Russia will increase 2 per cent next year and the uninflation rate increase from 5% to 8%.

Zuo Zhongmo, deputy secretary-general of the Conference, said, "This is not just an issue of agriculture. These reclaimed lands can serve the general development purposes of various sectors, including forestry, industry and tourism."	Zuo Zhongmo, Under-Secretary-General of the Conference, said: “This is not just an agriculture issue, these recovered lands can serve the overall development goals of various sectors, including forestry, industry and tourism.	Zuo Zhongmo, Deputy Secretary-General of the Conference, said, "It is not just an agriculture issue; the land recovered from them can serve the overall development goals of different sectors including forestry, industry and tourism."	It is Zuo Zhongmo, Under-Secretary-General of the Conference, who said: It is not just about agriculture; the reclaimed lands can be used for general development in various sectors, including forestry, industry and tourism."	The Assistant Secretary-General of the Conference, Zuo Zhongmo, said, It is not just about agriculture, as the reclaimed lands can be used for general development purposes from various sectors, including forestry, industry and tourism.
In the United States, California and other southern states were flooded at the beginning of this year, followed by a drought in many places in the south.	In the United States, California and other southern states were flooded early this year, followed by droughts in many parts of the South.	In the United States of America, CaliFORNA and other southern states were flooded early this year, followed by drought in many parts of the south.	In the United States, California and other southern states, floods occurred in early this year, followed by drought in many areas of the south.	Inondations in California, and other southern States in early this year, followed by droughts in many endroits in the south.
Meanwhile, the US Congress is discussing whether or not to approve the Protocol reached in the Kyoto Conference in Japan.	In the meantime, the United States Congress is discussing whether or not to approve the Protocol concluded at the Kyoto Conference in Japan.	Meanwhile, US Congress debates whether or not to accede to the Protocol agreed at the Kyoto Conference in Japan.	At the same time, the United States Congress is examining whether or not to approve the Protocol at the Kyoto Conference in Japan.	At the same time, the United States Congress is considering whether or not to approve the Protocol made at the Kyoto Conference in Japan.
The index would fall 3.9% if the sales of vehicles were not included.	The index would fall by 3.9 per cent if vehicle sales were not included.	The Index would fall 3.9% if the vehicle sales were not included, or 3.9% if the vehicle sales were excluded.	The index would decline by 3.9% if the vehicle sales were not included.	The index would decline by 3.9% if vehicles were not included in sales.
According to the company, in the coming five years, the company will make an additional investment of US$90 million, with an anticipated annual output value of US$300 million.	According to the company, over the next five years, it will make an additional investment of US$90 million, with a projected annual production value of US$300 million.	According to the company, it will make an additional $90 million US over the next five years with a planned annual productivity value of $300 million US.	According to the company, over the next five years it will invest an additional $90 million in the U.S., its expected annual output of $300 million.	According to the company, it will invest $90 million in the next five years, with its expected annual output of approximately $300 million.
Some people worry that this will affect the business of large-sized Canadian enterprises.	There are concerns that this may affect the operations of large Canadian companies.	There is concern that this may affect what large Canadian firms have in their business.	Some may be concerned that this will affect the major Canadian enterprises.	Some people worry that this will impact on the great enterprises in Canada.
A Majority of Hong Kong Residents Decline to Consider Themselves as Chinese	A majority of Hong Kong residents decline to regard themselves as Chinese	The Hong Kong City of Hong Kong’s minority people are not recognising themselves as Chinese	The vast majority of Hong Kong residents feel they are being referred as Chinese	The vast majority of Hong Kong residents have become disenfranchised as Chinese
He said the 83-year-old woman has been hospitalized for overshock.	He said that the 83-year-old woman had been hospitalized for overheating.	He said that the 83-year-old woman had been hospitalized for overheat.	He stated that a 83-year-old woman had been hospitalized for a headache.	He indicated that a 83-year-old woman had been hospitalized for a head injuries.
However, statistics released by the Immigration Bureau showed that although there were 11,978 new British immigrants coming to Australia between 1996 and 1997, 3,737 people left the country during the same	However, statistics published by the Immigration Bureau show that, although there were 11,978 new British immigrants to Australia between 1996 and 1997, 3,737 people left the country during the same	However, Immigration Bureau statistics show that although there were 11,978 new British immigrants to Australia from 1996 to 1997, 3,737 individuals left the country during that period.	However, the statistics of the Office of Immigration show that, although 11,978 new British immigrants arrived in Australia between 1996 and 1997, 3,737 had left the country during the same period	However, the figures from the Immigration Bureau show that, although 11,978 new British immigrants had entered Australia between 1996 and 1997, 3,737 had left the country during the same period.

(Reuters report from Tokyo)Japanese Finance Minister Kiichi Miyazawa was pressured not to quit office yesterday.	Japan’s Finance Minister, Kiichi Miyazawa, was forced yesterday not to leave his post.	The finance minister in Japan, Kiichi Miyazawa, was forced to leave the post yesterday.	(Tokyo report) Japan’s Minister of Finance Kiichi Mi Theawa was pressured not to quit yesterday.	(Tokyo report) The Japanese Finance Minister Kiichi Mi ichiawa was pressured on not to leave yesterday.
US Admitted Hundreds of Deaths of Iraqi Civilians in Air Strike and UK Reporter Claimed the Target Being Definitely Non-military	US Admitted Hundreds of Deaths of Iraqi Civilians in Air Strike and UK Reporter Claimed the Target Being Definitely Non-military	U.S. Admitted Hundreds of Deaths of Iraqi Civilians in Air Strike and UK Reporter Claimed the Target Being Definitely Non- Military	The US killed hundreds of Iraqi civilians in an air strike, and a reporter in the UK said the target was essentially non-military.	Several hundred dead of Iraqi civilians in air strikes; a UK reporter said that its specific goal is, essentially, non-military.
During the Eighth Five-Year Plan Period (from 1991 to 1995), township enterprises in Fujian Province contributed an aggregate total of RMB18.56 billion in tax, and achieved a total of RMB105.5 billion worth of export commodities.	During the eighth period of the five-year plan (1991-1995), municipal enterprises in Fujian Province paid taxes totalling RMB 18.56 billion and exported products totalling RMB 105.5 billion.	During the eighth period of the Five-Year Plan (from 1991 to 1995), Fujian Provincial Municipal Enterprises had paid taxes totalling RMB 18.56 billion and exported revenues totalling RMB 105.5 billion.	During the eighth five-year plan (from 1991 to 1995), businesses in the townships of Fujian Province contributed a total of $18.56 billion in tax contributions, $105.5 billion in commodities for export.	During the eighth five-year plan (from 1991 to 1995), businesses in the townships of Fujian Province contributed tax contributions totalling $118.56 billion in 1991-95, as well as $1005.5 billion in commodities for export.
In May this year, Dole admitted using Viagra on a trial basis, and gave high remarks to the drug after use, describing it as "a magic drug."	In May of this year, Dole admitted to using Viagra on an experimental basis and commented very positively on the drug after its use, describing it as "a magical drug."	In May this year, Dole admitted that he uses Viagra as an experimental patient and provided very positive feedback on the drug after its use describing it "a magical medicine".	In May of this year, Dole accepted the use of marijuana for trials and reported strong post-treatment remarks, describing it as a magical drug.	In May of this year, Dole recognized the use of pesticide in trials and had reported very good after-treatment, describing it as the “ magical drug”.
These people can set examples and they can have direct influence over the improvement of local human rights conditions and the protection of employees.	These individuals can provide examples and have a direct influence on the improvement of local human rights and employee protection conditions.	These may lead to examples and direct influence on the betterment of local human rights conditions and on the protection of wage earners.	They can provide examples and can directly influence the improvement of local human rights conditions and the protection of employees.	They can provide examples and have direct influence on improving local human rights and the protection of employees’ conditions.

Table 3: Sample paraphrases generated by the different methods.

PoeLM: A Meter- and Rhyme-Controllable Language Model for Unsupervised Poetry Generation

itor Ormazabal[1] Mikel Artetxe[2] Manex Agirrezabal[3] Aitor Soroa[1] Eneko Agirr
[1]HiTZ Center, University of the Basque Country (UPV/EHU)
[2]Meta AI [3]University of Copenhagen
{aitor.ormazabal,a.soroa,e.agirre}@ehu.eus
artetxe@meta.com manex.aguirrezabal@hum.ku.dk

Abstract

Formal verse poetry imposes strict constraints on the meter and rhyme scheme of poems. Most prior work on generating this type of poetry uses existing poems for supervision, which are difficult to obtain for most languages and poetic forms. In this work, we propose an unsupervised approach to generate poems that follow any given meter and rhyme scheme, without requiring any poetic text for training. Our method works by splitting a regular, non-poetic corpus into phrases, prepending control codes that describe the length and end rhyme of each phrase, and training a transformer language model in the augmented corpus. The transformer learns to link the structure descriptor with the control codes to the number of lines, their length and their end rhyme. During inference, we build control codes for the desired meter and rhyme scheme, and condition our language model on them to generate formal verse poetry. Experiments in Spanish and Basque show that our approach is able to generate valid poems, which are often comparable in quality to those written by humans.

1 Introduction

Despite the impressive generative capabilities of large Language Models (LMs) (Brown et al., 2020; Chowdhery et al., 2022; Zhang et al., 2022) automatic poetry generation remains a challenging problem. **Formal verse poetry**, in particular, imposes strict constraints on the meter and rhyme scheme of poems (Figure 1), which cannot be directly controlled in conventional LMs.

Prior work on generating formal verse poetry has primarily focused on supervised approaches, leveraging existing poems to train LMs. This is often combined with additional techniques to impose the meter and rhyme constraints at inference time, such as using finite-state automata to discard invalid candidates (Ghazvininejad et al., 2016), or generating text right-to-left to better control the rhyming word (Lau et al., 2018; Jhamtani et al., 2019; Xue et al., 2021). However, these approaches require poetic text for training, which is difficult to obtain for most languages and poetic forms.

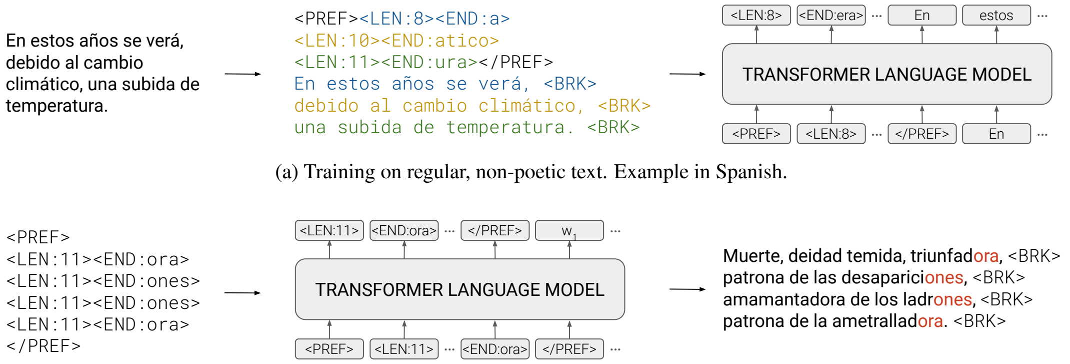
```
   Pen|san|do | que el | ca|mi|no i|ba | de|re|cho, →<LEN:11><END:echo>
     vi|ne a | pa|rar | en | tan|ta | des|ven|tu|ra,  →<LEN:11><END:ura>
que i|ma|gi|nar | no | pue|do, aún | con | lo|cu|ra,→<LEN:11><END:ura>
   al|go | de | que es|té un | ra|to | sa|tis|fe|cho.  →<LEN:11><END:echo>
```

Figure 1: **A formal verse poem and its associated structure descriptor.** The poem is the first stanza of a Spanish sonnet, which must have 4 lines of 11 syllables and follow an ABBA rhyme scheme. We use control codes to describe such constraints, and train a language model that can generate text conditioned on them.

In this paper, we propose an unsupervised approach to generate formal verse poetry. Our **Poe**tic **L**anguage **M**odel (PoeLM) can be conditioned to follow any desired meter and rhyme scheme, without requiring any poem for training. As illustrated in Figure 2, the key idea behind our method is that any text can be divided into phrases, which will each have a certain number of syllables and end in a certain sound that can make it rhyme with other phrases. While this structure will not follow a regular pattern for standard text, as it would for poetry, we can still annotate it automatically, and train a language model that can be conditioned on such structure descriptors. At inference time, we build a structure descriptor for the desired meter and rhyme scheme, and condition our language model on it to generate formal verse poetry. To improve results, we generate multiple candidates, which are automatically filtered and re-ranked.

Our experiments in Spanish and Basque show that our method is able to generate high quality poems meeting the desired meter and rhyme constraints, with human evaluators ranking our system higher than other humans in more than one third of the cases. Our code is available at GitHub.[1]

[1] https://github.com/aitorormazabal/poetry_generation


En estos años se verá, debido al cambio climático, una subida de temperatura.
<PREF><LEN:8><END:a>
<LEN:10><END:atico>
<LEN:11><END:ura></PREF>
En estos años se verá, <BRK>
debido al cambio climático, <BRK>
una subida de temperatura. <BRK>
<LEN:8> <END:era> ··· En estos ···
TRANSFORMER LANGUAGE MODEL
<PREF> <LEN:8> ··· </PREF> En ···
<PREF>
<LEN:11><END:ora>
<LEN:11><END:ones>
<LEN:11><END:ones>
<LEN:11><END:ora>
</PREF>
<LEN:11> <END:ora> ··· </PREF> w1 ···
TRANSFORMER LANGUAGE MODEL
<PREF> <LEN:11> ··· <END:ora> </PREF> ···
Muerte, deidad temida, triunfadora, <BRK>
patrona de las desapariciones, <BRK>
amamantadora de los ladrones, <BRK>
patrona de la ametralladora. <BRK>

(a) Training on regular, non-poetic text. Example in Spanish.

(b) Generation of formal verse poetry. Example in Spanish.

Figure 2: **Proposed method.** (a) During training, we split non-poetic text into phrases according to punctuation marks, prepend control codes describing the length and end rhyme of each phrase, and train a transformer language model on it. (b) During inference, we build a structure descriptor with control codes for the desired meter and rhyme scheme, and condition our language model on them to generate formal verse poetry.

2 Background: formal verse poetry

Poetic traditions differ across languages and cultures. In this work, we focus on **formal verse poetry** in Spanish and Basque,[2] which impose strict meter and rhyme constraints as follows:

- The **syllabic meter** specifies the number of lines in the poem, as well as the number of syllables that each line must contain.[3] Spanish syllabic meter allows for synalephas, where two syllables can be merged into one when one word ends in a vowel and the next starts with one. For simplicity, we do not consider synalephas when counting syllables, although our method could easily be extended to account for them.

- The **rhyme scheme** specifies the pattern according to which lines must rhyme. For instance, the ABAB scheme requires the 1st line to rhyme with the 3rd one, and the 2nd line to rhyme with the 4th one. Two lines are considered to rhyme if they repeat the same sound at their last syllables.[4] In addition, rhyming lines cannot end in the same word.

There are different **poetic forms** depending on the specific meter and rhyme scheme that they impose. For instance, the first stanza of a Spanish sonnet must consist of four verses with 11 syllables each, following an ABBA rhyme scheme. As illustrated in Figure 1, we use control codes to define such meter and rhyme constraints, which we refer to as **structure descriptors**.

3 Proposed method

As described in §2, we want our system to be able to generate text that adheres to a specific structure. The key idea behind our approach is that, similar to formal verse poetry, any text adheres to a certain implicit structure. In the case of non-poetic text the structure will not follow any regular pattern, but we can still extract it and build a structure descriptor for it. We can then augment the non-poetic corpus with these descriptors, and train a regular LM on it (Figure 2a). The model thus learns to respect the structure provided in the descriptor, which allows us to generate formal verse poetry at inference time, by conditioning the model on the appropriate structure descriptor (Figure 2b).

We next describe the two main components of our method: structure-aware training (§3.1) and inference with filtered re-ranking (§3.2).

[2] The selected languages where narrowed down according to the availability of publicly available high-quality syllabization and rhyme detection systems (which discarded English), as well as the fluency of the authors.

[3] Some traditions impose a stress pattern in addition to the number of syllables, which is known as *accentual-syllabic meter*. We do not consider this type of meter in our work, as it is not common in Spanish and Basque.

[4] In Spanish, two words rhyme if their sounds are identical from the last stressed vowel onwards. In Basque, two words rhyme if their sounds match from the first vowel of the second to last syllable onwards, and the following consonant groups are considered to sound the same for the purposes of rhyme: {p,t,k}, {n,m}, {s,z,x}, and {b,d,g,r}.

3.1 Structure-aware training

Let X be the space of possible text sequences, and S be the space of possible structure descriptors. We can define a function $s : X \to \mathcal{P}(S)$ that maps each sequence of text into its corresponding set of descriptors.[5] We want to build a model that can sample from $P(X|c \in s(X))$, that is, that can sample text conditioned on an structure descriptor c. In theory, one could do this through rejection sampling, by repeatedly drawing sentences from $x \sim P(X)$ until one of them satisfies $c \in s(x)$. However, this is intractable in practice, since the probability of a randomly sampled text following the desired structure is practically zero.

Instead, we train a LM that can be conditioned on any given structure (see Figure 2a). To that end, we start by **annotating the implicit structure** of a regular, non-poetic corpus. We first split the corpus in phrases, where we define a phrase as a sequence of text delimited by either a newline or a punctuation character (e.g., commas, colons or quotes). We do this so that the rhyme words at the end of these units correspond to natural stopping points. We then group the text in blocks of n phrases, where n is randomly sampled. For each block x, we choose a structure descriptor $c_x \in s(x)$ that defines the length and end rhyme of each of the phrases it contains. We then **create an augmented corpus** $(c_{x_1}, x_1, c_{x_2}, x_2, ...)$ by interleaving the previously generated structure descriptor c_{x_i} before its corresponding text block x_i (see Appendix A for more details). Finally, we **train a transformer LM** on the augmented corpus. The control codes in the structure descriptors are treated as regular tokens, and the model is trained with the standard next token prediction objective.

3.2 Generation with filtered re-ranking

At inference time, we use the LM from §3.1 to generate formal verse poetry in 3 steps:

1. Candidate generation. We specify the desired meter and rhyme scheme as a structure descriptor,[6] and use our LM to generate text conditioned on it (see Figure 2b). We repeat the process $k = 3000$ times to generate k different candidates. In our experiments, we provide the first line of the poem to generate in addition to its structure descriptor, which is useful to define the subject and make different systems easier to compare.

2. Filtering. In practice, some of the generated candidates do not meet the given constraints or are otherwise pathological. For that reason, we filter candidates according to the following conditions:

1. **#Line.** The candidate must have the number of lines specified in the structure descriptor.
2. **#Slb.** Each line must have the number of syllables specified in the structure descriptor.
3. **Rhyme.** Each line must end in the rhyme sound specified in the structure descriptor.
4. **Rep. word.** No two rhyming lines can end in the same word.
5. **BLEU.** In order to prevent the model from generating repetitive text, the maximum and average BLEU across any two lines must be be less than or equal to 35 and 20.

3. Re-ranking. We score the remaining candidates for fluency using our LM, and output the one with the highest score. Different from the first step, we do not condition on the structure descriptor when doing so, which gives a measure of the general fluency.

We test the efficacy of the second and third steps in the experiments.

4 Experimental design

We run experiments on Spanish and Basque. We next detail the training details (§4.1) and the automatic and human evaluation setup (§4.2 and §4.3).

4.1 Training details

Hyperparameters. We train transformer LMs using the same settings as Brown et al. (2020). For Basque, we train a 350M model with a learning rate of 3×10^{-4} and linear decay over 300B tokens.[7] For Spanish, we train a 760M model over 100B tokens using a constant[8] learning rate of 2.5×10^{-4}.

[5] Each sequence is mapped to a subset of S, as the same sequence could be described by multiple descriptors.

[6] A rhyme scheme specifies which lines must rhyme, but not what the rhyme sound should be. We thus generate a concrete structure descriptor from the given scheme by sampling each rhyme sound independently from the five most common rhyme sounds in the training corpus.

[7] In practice, we stop training after seeing around 85B tokens, when performance plateaus in the validation set.

[8] We initially planned to manually decay the learning rate according to validation perplexity. However, we did not observe performance plateauing (presumably due to the large corpus and our constrained compute budget), so the full training was done with a constant learning rate.

Corpora. We use EusCrawl (Artetxe et al., 2022) as our training corpus for Basque, which takes 2.5GB in plain text format, and a subset of 700GB from mC4 (Raffel et al., 2019) for Spanish. Given the small size of the Basque corpus, we combine 10 versions of the corpus using different random seeds to generate the structure descriptors.

Preprocessing. We use SentencePiece tokenization (Kudo and Richardson, 2018) with a 50k vocabulary for each language, and reserve 8.5k tokens for the control codes in the structure descriptors. For syllabification and rhyme sound extraction we use the rules provided by Agirrezabal et al. (2012),[9] which are encoded as finite-state transducers implemented in Foma (Hulden, 2009).

Models. In addition to our proposed model (PoeLM), we train a regular LM for each language as a baseline, using the exact same hyperparameters, tokenization, and corpora (without the interleaved structure descriptors).

4.2 Automatic evaluation

We use Spanish poems from the 20th century subset of the DISCO dataset (Barbado et al., 2021), and Basque poems from the BDB dataset[10] to evaluate our approach. The DISCO and BDB datasets consist of 20k and 44k tokens before our SentencePiece tokenizer is applied, respectively. We use the following automatic metrics:

Filtering rate. We take 10 poems[11] from each test set, extract the first line from them, generate poems for each as described in §3.2 following the meter and rhyme scheme of the original poem, with $k = 3000$ candidates for each, and measure the percentage of candidates that are filtered according to the criteria in §3.2. We compare the resulting filtering rate of our proposed PoeLM, which is conditioned on the relevant structure descriptor, and a regular LM, which is not conditioned on any structure but could still generate a valid poem given enough trials.

Since, unlike PoeLM, the baseline LM does not generate break tokens to separate lines, we split the generated text into lines according to the relevant number of syllables. When this cannot be done while respecting word boundaries, we consider that the candidate is rejected for breaking the *#slb* condition. As a consequence, generations from the baseline LM are never considered to be rejected due to the *#verse* condition.

Perplexity. To understand how well the model is able to leverage the information provided by a known structure, we compare the per-token perplexity of (i) PoeLM conditioned on the relevant structure descriptor, (ii) PoeLM without conditioning on any structure descriptor, (iii) the baseline LM. We do this both in the validation set of the non-poetic corpus used for training, as well as the poem datasets used for evaluation.

Consistent with training, we insert break tokens to separate lines for both PoeLM variants. However, these special tokens are excluded from the perplexity computation to make them comparable with the baseline LM.

4.3 Human evaluation

We run a qualitative evaluation in Spanish comparing poems generated by our system and humans. Given that writing poems is also challenging for humans, we consider both poems written by actual poets as well as layman volunteers. More concretely, we take the first line of 50 poems from the DISCO dataset, and compare 3 poems generated by completing them as follows:

- **Expert**: The original poem from DISCO from which the first line was extracted, authored by a renowned poet.
- **Layman**: Poems written by non-expert volunteers within a time limit of about 5 minutes.
- **PoeLM**: Poems generated by our system using the full pipeline described in §3.2.

We then give these 3 poems[12] to human evaluators in a random order, and ask them to rank from best to worst. We report results according to two metrics: the overall rank (the percentage of times that each system has been ranked in each position), and the head-to-head comparison (the percentage

[9] https://bitbucket.org/manexagirrezabal/syllabification_gold_standard

[10] https://bdb.bertsozale.eus/. We use the 2005 segment of the corpus.

[11] For Spanish, we use the first Stanza of full sonnets from DISCO, which consist of either 11 or 14 syllable lines, following a rhyme scheme of ABAB or ABBA. For Basque, we use *Zortziko Handia* poems from BDB, which consist of 8 lines, where the odd ones are 10 syllables long, the even ones are 8 syllables long, and only the even lines are required to rhyme.

[12] A 4th candidate, which we ignore when calculating the ratings, was also included for the analysis in §6.2.

	Spanish		Basque	
	PoeLM	LM	PoeLM	LM
Correct	30.9	0.0	23.4	0.0
Reject due to				
#Verse	3.7	0.0	9.6	0.0
#Slb	17.0	96.6	34.0	90.3
Rhyme	13.1	3.4	11.1	9.7
Rep. word	31.1	-	19.7	-
BLEU	4.2	-	2.3	-

Table 1: Percentage of filtered candidates, with a breakdown for the reason of rejection. See §4.2 for details.

		Spanish		Basque	
		poetic	prose	poetic	prose
Baseline LM		62.7	15.9	151.1	24.3
PoeLM	w/ struc	49.5	11.7	42.5	10.1
	no struc	129.5	18.0	634.2	81.4

Table 2: Perplexity of poetic and non-poetic (prose) corpora. See §4.2 for details.

of times that each system has been ranked before each other system).

All volunteers that wrote the poems, as well as those that ranked the candidates, are native Spanish speakers with university studies. While there was an overlap between both groups of volunteers, we made sure that volunteers were never asked to rank poems written by themselves. All volunteers are familiar with the fundamentals of formal verse poetry, but are not experts in the matter. Refer to Appendix B for more details.

5 Results

We next discuss our main results for the automatic (§5.1) and human evaluation (§5.2).

5.1 Automatic evaluation

We report **filtering rate** results in Table 1. We find that 30.9% of Spanish poems and 23.4% of Basque poems sampled from PoeLM meet the given constraints. While far from perfect, this means that sampling a few candidates is enough to obtain a valid poem with our approach. In contrast, none of the poems generated by the baseline LM is valid, showing that our proposed structure-aware training is critical to generate formal verse poetry with LMs. Regarding the reason for rejection, we find that the majority of candidates from PoeLM are discarded for repeating rhyming words, which the model was not directly trained to prevent.

S2 S1	Expert	Layman	PoeLM
Expert	-	54.0%	62.7%
Layman	46.0%	-	60.7%
PoeLM	37.3%	39.3%	-

Table 3: Percentage of times that system $S1$ is ranked ahead of $S2$ in the human evaluation.

	1st	2nd	3rd
Expert	44.0%	28.6%	27.3%
Layman	36.7%	33.3%	30.0%
PoeLM	19.3%	38.0%	42.7%

Table 4: Percentage of times that each system has been ranked in each position in the human evaluation.

Table 2 reports the **perplexity** results. When conditioned on structure descriptors, our model always outperforms the baseline LM, meaning that it is able to make better predictions accounting for the meter and rhyme constraints. However, when the structure descriptor is not provided, our model's perplexity is higher, presumably because the model did not see text without structure descriptors during training.

5.2 Human evaluation

We report head-to-head results in Table 3, and ranking results in Table 4. Human evaluators prefer poems generated by our system over those written by renowned poets in 37.3% of the cases. Similarly, our system does better than laymen in 39.3% of the cases. This shows that our system is able to generate high-quality poems, which humans often prefer over poems written by other humans. This can also be seen in the ranking evaluation, as our system has been ranked in first position in 19.3% of cases, and among the first two positions in 57.3% of cases.

Finally, it is surprising that layman poems are ranked above those from renowned poets nearly half of the times. We attribute this to the human evaluators themselves being laymen, leading them to prefer poems that use more plain language. This is also reflective of the subjective nature of the task, as different readers might enjoy poetry differently.

6 Analysis

We further analyze our system by quantifying at which portion of the poem its perplexity gain is highest (§6.1), experimenting with manual re-

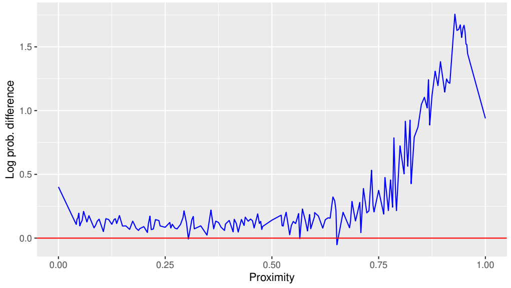


Figure 3: Interpolated advantage in log probability of our model compared to a regular LM over the Spanish mC4 validation set, as a function of normalized proximity to the next specified rhyme token. See §6.1 for details.

S2 / S1	Exp.	Lay.	PoeLM	PoeLM +*rerank*
PoeLM	37.3	39.3	-	26.0
PoeLM +*rerank*	41.3	42.7	38.0	-

Table 5: Percentage of times that system $S1$ is ranked ahead of $S2$ in the human evaluation. Since the candidate chosen by a human annotator among the top 6 candidates will sometimes be the same as the top candidate, there can be ties, and thus the head-to-head percentages do not add up to 100.

ranking (§6.2), and looking at some sample poems (§6.3).

6.1 Perplexity gain

We quantify the predictive advantage of our system as a function of the distance to the next rhyme word. To this end, we plot the difference in token-wise log probabilities between our model and the baseline LM as a function of proximity to the next rhyme word, interpolated between 0 and 1. We only consider lines with 15 to 25 tokens.

As shown in Figure 3, our model's advantage is greatest near the rhyme word. This is not surprising, as there is less uncertainty towards the end of the line when the meter and rhyme are known. We observe a downward spike towards the end, that may initially seem counter-intuitive. We hypothesize that, since the rhyme word will often be split into multiple tokens, by the time the first tokens of the rhyme word are known the regular LM will be quite sure of what the word is, meaning that the advantage of knowing the rhyme is lower.

6.2 Manual re-ranking

A potential application of automatic poetry generation is helping (rather than replacing) humans when writing poems. As a first approximation, we ask our volunteers to manually choose a poem among the top 6 candidates generated by our system.[13] The resulting poem was considered as part of the human evaluation described in §4.3, and compared to the other 3 systems.

Table 5 reports the head-to-head performance of our model with and without manual re-ranking. As expected, the re-ranked model performs better, beating the poems generated by laymen in 42.7% of cases, as opposed to 39.3% for the base system. However, the base system beats manual re-ranking in 26% of cases, meaning that human evaluators often prefer the top candidate automatically selected by the system over the one manually selected by another human. This means that there is a considerable disagreement across annotators, which is reflective of the subjective nature of the task.

6.3 Sample poems

Tables 6 and 7 show some example poems generated by our system in Spanish and Basque. The poems were generated by providing the first line along with the structure descriptor to the system, and manually selecting a candidate among the top six. Five lines were selected at random from the evaluation in Spanish, and two for Basque. The full list with the six candidates is given in Appendix C. No *cherry-picking* was done, except to choose one poem among the six candidates per line.

We observe that the system is capable of generating coherent poems covering varied topics. For example, regarding the Spanish poems, the first, third and fourth Spanish poems cover themes of inner conflict, the second one describes a person's beauty, and the last is about an abbey called Monserrat. Note that the theme is implicit in the first line, and mirrors the typical topics of Spanish sonnets of the time. Regarding the Basque poems, the themes are friendship and swings in a park, also mirroring the themes used in contemporary spontaneous poetry contests in the dataset.

7 Related work

We next review relevant literature in poetry generation (§7.1), as well as controllable generation

[13] We take the top three candidates with and without BLEU filtering to generate this list.

Generated Poem	English translation
Siento otro Yo que contra mí se empeña, un Yo para el que no debo luchar, un Yo al que no debo acompañar, un Yo que a menudo me condena.	I feel another Self that is set against me, a Self for which I must not fight, a Self that I should not accompany, a Self that often condemns me.
Casta su faz, bajo la blanca toca, su faz es dulce,es bella sin cesar, su faz es hermosa como un jaguar, su faz es divina como la roca.	Chaste is her face, under her white cap, her face is sweet, relentlessly gorgeous, her face is beautiful like a jaguar, her face is divine like rock.
Nada más. De aquel sueño, que pasó como un ala, arranco todo lo que había en mi mente, todo lo que me atormentaba y no cala, lo que callo en el interior de mi presente.	Nothing more. Of that dream, that passed like a wing, it tore everything from my mind, all that tormented me and doesn't seep through, which I keep quiet inside my present.
Es inútil que luche por quitarme del pecho, que niegue repetidamente mis opiniones, que trague de nuevo mi entusiasmo deshecho, que rechace de nuevo todas mis negaciones.	Fighting to get it off my chest is futile, that I repeatedly deny my emotions, that I once again swallow my undone enthusiasm, that I once again reject all my negations.
Del Monserrat en la penumbra undosa, Del Monserrat silente en el Solar, Del Monserrat dolido en el remar, Del Monserrat cautivo en la prosa.	Of the Monserrat in the gloomy twilight, Of the Monserrat, silent in sunlight, Of the Monserrat, pained in paddling, Of the Monserrat, captive in prose.

Table 6: Spanish poems generated by our method, given five lines selected at random from the dataset. The five poems have been manually selected from the top six candidates generated by the system for each line, with no other form of *cherry-picking*. See Appendix C for the full list of six candidate poems.

(§7.2).

7.1 Poetry generation

Retrieval based approaches. Early work in poetry generation focused on rule-base methods, which generate text according to predefined rules that ensure the desired structure is followed (Gervás, 2000; Gonçalo Oliveira et al., 2007). A popular approach is to fill templates with text extracted from existing poems (Colton et al., 2012; Gonçalo Oliveira, 2012; Gonçalo Oliveira et al., 2017). This makes it easy to control poetic structure, since the meter and rhyme schemes of the text pieces can be annotated in advance and combined accordingly when filling the templates. However, the diversity and creativity of these approaches is limited.

Neural poetry generation. More recently, there has been work on applying neural text generation to poetry. A popular approach is to train a finite-state acceptor (FSA) that ensures all accepted sequences obey the required structure, which is then used to guide a recurrent neural network (RNN) through rejection sampling (Ghazvininejad et al., 2016, 2018; Hopkins and Kiela, 2017). However, these methods require some form of lyrical or poetic text to train the RNN or the FSA, and they must generate text right-to-left in order to respect rhyme sounds, as the model has no concept of planning. Additionally, a new FSA has to be trained for each desired poem structure. Lau et al. (2018) augment an RNN with a pentameter model and learn the meter and rhyme constraints of sonnets in a supervised way from a sonnet corpus. They then generate poem lines right-to-left, to alleviate the model's lack of planning. Van de Cruys (2020) trains an encoder-decoder RNN on prosaic text to generate each line right-to-left conditioned on the previous one, and applies constraints when decoding to ensure the generated text adheres to a rhyme scheme and consistent topic. However, their system cannot enforce a specific syllabic meter.

Multiple works focus on neural poetry generation for the Chinese language, applying techniques such as reinforcement learning (Yi et al., 2018) or planning (Wang et al., 2016). In Chinese, one character corresponds to a syllable, but meter is governed by tonal constraints. Most of the reviewed works assume that, with a sufficiently large corpus, the model should be able to learn the implicit tonal structure of poetry (Wang et al., 2016; Zhang et al., 2017; Liu et al., 2018). Yeh et al. (2019) concatenate tonal information to the character embeddings of an LSTM to create a model that is more phonologically compliant.

Notably, current neural methods capable of controlling both syllable count and rhyme scheme require some form of poetic corpus to train, and usu-

Generated Poem	English translation
Gu biok lagun handiak gara, anaia,aita,semea, eta bion ideologia, gure identitatea, konpartitzen dugu.Batzuetan, zaila da bat esatea, besteak ulertzea,benetan, zein ahula den bestea.	The both of us are great friends, brother,father,son, and our ideology, our identity, is shared. Sometimes, it is hard to say one, to understand others, truly, how weak others are.
Nahiz kulunpio pila bat egon, eguzkiak sikiera, aukera du ondo goxatzeko, eta ez beti gainera, baita asteko egun denetan, baita hemendik aurrera, ilargi erdiko orduetan, eta hori da ederra.	Even though there are many swings, at least the sun, has a chance to enjoy, not always, also during every day of the week, and, from now on, during the moon hours, and that is beautiful.

Table 7: Basque poems generated by our method, given two lines selected at random from the dataset. The poems have been manually selected from the top six candidates generated by the system for each line, with no other from of *cherry-picking*. See Appendix C for the full list of six candidate poems.

ally generate text right-to-left to alleviate a lack of planning when generating rhymes.

7.2 Controllable generation

Similar to our approach, several works attempt to control the generated output by augmenting the training data with tags. Keskar et al. (2019) augment the training corpus of a LM with codes automatically extracted from metadata. Some works in machine translation explore augmenting the training data in order to control the politeness (Sennrich et al., 2016), domain (Kobus et al., 2016), or length (Lakew et al., 2019) of generated translations. Schioppa et al. (2021) experiment with vector-valued additive tags in order to control multiple attributes of the generated text at once. However, all of these systems use tags that only broadly specify the length, domain or style of the text to generate. In contrast, our model is conditioned on a very specific meter and rhyme scheme that the text must follow.

8 Conclusions and future work

In this work, we present an unsupervised approach to generate formal verse poetry. We identify and extract the latent structure in non-poetic corpora, and feed this information along with the text to a transformer LM, allowing us to control the structure of the text at generation time. Our system is capable of generating formal verse poetry with flexible meter and rhyme schemes, without requiring any sort of poetic text to train. The required structure can be easily altered by changing the descriptor, allowing us to generate different types of poetry without needing to re-train the system. Automatic and human evaluations show that our model learns to leverage the provided structure information to better predict the text, and is capable of generating short poems that are often preferred to those created by a human.

In future work, we would like to extend our framework to be able to control other aspects of the generated text in addition to meter and rhyme.

Limitations

Given that our method requires tagging the implicit meter and rhyme of the training corpus, we are limited by the quality of available syllabization and rhyme detection systems. While rule-based systems with a low error rate are easy to create for languages such as Spanish or Basque, this is not the case for English, which is why we did not train an English version of our system. However, our approach is independent of the used syllabization and rhyme detection process, and could be readily applied on top of any system with a low error-rate.

Additionally, our Spanish syllabization system has no concept of synalephas, where two syllables can be merged into one when one word ends in a vowel and the next starts with one. This means that our system will never use this Spanish literary device when generating poems.

Acknowledgements

Aitor Ormazabal, Aitor Soroa and Eneko Agirre were partially supported by the Basque Government (IXA excellence research group IT1343-19). Aitor was supported by a doctoral grant from the Spanish MECD.

We would like to thank Ainara Estarrona, Begoña Altuna and Itziar Gonzalez-Dios for their assistance in defining literary terminology, and Edward Yao for his help translating poems.

References

Manex Agirrezabal, Inaki Alegria, Bertol Arrieta, and Mans Hulden. 2012. Finite-state technology in a verse-making tool. In *Proceedings of the 10th International Workshop on Finite State Methods and Natural Language Processing*, pages 35–39.

Mikel Artetxe, Itziar Aldabe, Rodrigo Agerri, Olatz Perez de Viñaspre, and Aitor Soroa. 2022. Does corpus quality really matter for low-resource languages?

Alberto Barbado, Víctor Fresno, Ángeles Manjarrés Riesco, and Salvador Ros. 2021. DISCO PAL: Diachronic spanish sonnet corpus with psychological and affective labels. *Language Resources and Evaluation*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. Palm: Scaling language modeling with pathways.

Simon Colton, Jacob Goodwin, and Tony Veale. 2012. Full-face poetry generation. In *International Conference on Computational Creativity 2012*, pages 95–102. University College Dublin.

Pablo Gervás. 2000. Wasp: Evaluation of different strategies for the automatic generation of spanish verse. In *Proceedings of the AISB-00 symposium on creative & cultural aspects of AI*, pages 93–100.

Marjan Ghazvininejad, Yejin Choi, and Kevin Knight. 2018. Neural poetry translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 67–71, New Orleans, Louisiana. Association for Computational Linguistics.

Marjan Ghazvininejad, Xing Shi, Yejin Choi, and Kevin Knight. 2016. Generating topical poetry. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1183–1191, Austin, Texas. Association for Computational Linguistics.

Hugo Gonçalo Oliveira. 2012. Poetryme: a versatile platform for poetry generation. *Computational Creativity, Concept Invention, and General Intelligence*, 1:21.

Hugo Gonçalo Oliveira, Amílcar Cardoso, and Francisco Pereira. 2007. Exploring different strategies for the automatic generation of song lyrics with tra-la-lyrics. In *Proceedings of 13th Portuguese Conference on Artificial Intelligence, EPIA*, pages 57–68.

Hugo Gonçalo Oliveira, Raquel Hervás, Alberto Díaz, and Pablo Gervás. 2017. Multilingual extension and evaluation of a poetry generator. *Natural Language Engineering*, 23(6):929–967.

Jack Hopkins and Douwe Kiela. 2017. Automatically generating rhythmic verse with neural networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 168–178, Vancouver, Canada. Association for Computational Linguistics.

Mans Hulden. 2009. Foma: a finite-state compiler and library. In *Proceedings of the Demonstrations Session at EACL 2009*, pages 29–32.

Harsh Jhamtani, Sanket Vaibhav Mehta, Jaime Carbonell, and Taylor Berg-Kirkpatrick. 2019. Learning rhyming constraints using structured adversaries. In *Proceedings of the 2019 Conference on Empirical*

Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 6025–6031, Hong Kong, China. Association for Computational Linguistics.

Nitish Shirish Keskar, Bryan McCann, Lav Varshney, Caiming Xiong, and Richard Socher. 2019. CTRL - A Conditional Transformer Language Model for Controllable Generation. *arXiv preprint arXiv:1909.05858*.

Catherine Kobus, Josep Maria Crego, and Jean Senellart. 2016. Domain control for neural machine translation. *CoRR*, abs/1612.06140.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Surafel Melaku Lakew, Mattia Di Gangi, and Marcello Federico. 2019. Controlling the output length of neural machine translation. In *Proceedings of the 16th International Conference on Spoken Language Translation*, Hong Kong. Association for Computational Linguistics.

Jey Han Lau, Trevor Cohn, Timothy Baldwin, Julian Brooke, and Adam Hammond. 2018. Deep-speare: A joint neural model of poetic language, meter and rhyme. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1948–1958, Melbourne, Australia. Association for Computational Linguistics.

Dayiheng Liu, Quan Guo, Wubo Li, and Jiancheng Lv. 2018. A multi-modal chinese poetry generation model. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv e-prints*.

Andrea Schioppa, David Vilar, Artem Sokolov, and Katja Filippova. 2021. Controlling machine translation for multiple attributes with additive interventions. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6676–6696, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Controlling politeness in neural machine translation via side constraints. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 35–40, San Diego, California. Association for Computational Linguistics.

Tim Van de Cruys. 2020. Automatic poetry generation from prosaic text. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2471–2480, Online. Association for Computational Linguistics.

Zhe Wang, Wei He, Hua Wu, Haiyang Wu, Wei Li, Haifeng Wang, and Enhong Chen. 2016. Chinese poetry generation with planning based neural network. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1051–1060, Osaka, Japan. The COLING 2016 Organizing Committee.

Lanqing Xue, Kaitao Song, Duocai Wu, Xu Tan, Nevin L. Zhang, Tao Qin, Wei-Qiang Zhang, and Tie-Yan Liu. 2021. DeepRapper: Neural rap generation with rhyme and rhythm modeling. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 69–81, Online. Association for Computational Linguistics.

Wen-Chao Yeh, Yung-Chun Chang, Yu-Hsuan Li, and Wei-Chieh Chang. 2019. Rhyming knowledge-aware deep neural network for chinese poetry generation. In *2019 International Conference on Machine Learning and Cybernetics (ICMLC)*, pages 1–6.

Xiaoyuan Yi, Maosong Sun, Ruoyu Li, and Wenhao Li. 2018. Automatic poetry generation with mutual reinforcement learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3143–3153, Brussels, Belgium. Association for Computational Linguistics.

Jiyuan Zhang, Yang Feng, Dong Wang, Yang Wang, Andrew Abel, Shiyue Zhang, and Andi Zhang. 2017. Flexible and creative Chinese poetry generation using neural memory. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1364–1373, Vancouver, Canada. Association for Computational Linguistics.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. OPT: Open pre-trained transformer language models.

A Structure descriptors

The process of extracting the meter descriptors from a regular corpus and creating the augmented corpus consists of four steps:

1. First, we split the text into phrases according to the following set delimiters: _-?"!,:’‘()[].{}‘;»«><’. We do this so that the phrases, which will correspond to lines in

our generated poems, end at natural stopping points in speech. Additionally, we randomly merge each phrase with the next or the next two phrases, with probabilities 0.15 and 0.05, respectively, so that the model can generate verses that contain these special characters.

2. Second, we syllabize each phrase and extract the rhyme class of its final word, using our FOMA transducers.

3. Third, we split the text into blocks of n phrases, where n is sampled uniformly between 3 and 10. For each block, we construct a meter descriptor from the syllable count and rhyme class of each phrase. The descriptor begins with a `<PREF>` token and ends with a `</PREF>` token, and a pair of tokens of the form `<LEN_X> <CLS_Y>` for each phrase, where X is the syllable count and Y is the rhyme class. In 15% of cases, the rhyme class is replaced with a special `<CLS_UNK>` class, which allows us to leave the rhyme of certain verses unspecified when generating. Additionally, when there is a paragraph boundary (line break) in the text, we insert a `<SEP>` token in the corresponding position.

4. Fourth, we construct the corpus interleaving the meter descriptors in between the corresponding blocks of text. Additionally, we insert a `<BRK>` token in between phrases in the actual text. The `<BRK>` token lets us clearly see where the model has intended to end a phrase, allowing us to split the generated poems into verses easily.

B Human evaluation

As described in §4.3, we extract the first line from fifty real poems in the DISCO dataset, and compare the poems obtained by completing them. Specifically, for each case, we present the four candidate completions in a randomized order, and ask a human volunteer evaluator to rank them from best to worst. Additionally, the rhyme scheme that the poem should follow is provided, so that the evaluator can take it into account when judging a poem (our systems' poems and the gold standard never contain rhyme mistakes, but the poems generated by human annotators might). Each ranking is done by three different annotators.

All volunteers that participated in writing and ranking poems are familiar with the fundamentals of formal verse poetry, but are not experts in the matter. To obtain the "Layman" poems, volunteers were asked to complete a poem within a time limit of 5 minutes. However, this time limit was not enforced, and it was slightly exceeded in some cases.

C Full list of poems

Tables 8 and 9 show the full list of the six candidate poems generated by our system for each of the five lines in Spanish and two lines in Basque chosen at random from their respective datasets. No cherry-picking was done on these tables. From these poems, we did select one for each line manually, as shown in Tables 7 and 6 in the main body of the paper (see §6.3).

Siento otro Yo que contra mí se empeña, Siento otro Yo que no quiere cambiar, Siento otro Yo que no sabe amar, Siento otro Yo que no se condena.
Siento otro Yo que contra mí se empeña, siento otro Yo que no quiero amar, siento otro Yo que no puedo hablar, siento otro Yo que no me enseña.
Siento otro Yo que contra mí se empeña, y otro Yo que me quiere dominar, y otro Yo que no me quiere amar, que me condena y que me condena.
Siento otro Yo que contra mí se empeña, un Yo para el que no debo luchar, un Yo al que no debo acompañar, un Yo que a menudo me condena.
Siento otro Yo que contra mí se empeña, o una voz que me hace suspirar, o una risa que me hace llorar, o un grito que me quita la pena.
Siento otro Yo que contra mí se empeña, un Yo que no me deja amamantar, un Yo que me ensena a no llorar, un Yo que me enseña lo que pena.

Casta su faz, bajo la blanca toca, vestida de dos trpitos de azahar, Casta su faz,bajo la blanca tocar, vestida de dos trpitos de su boca.
Casta su faz, bajo la blanca toca, carvaja y danseá del palomar, Casta su faz,bajo la blanca tocar, carvaja y danseá de la roca.
Casta su faz, bajo la blanca toca, casta su faz sin tabiques que tapar, casta su faz cuando quiere cabalgar, casta su faz con guarnición de boca.
Casta su faz, bajo la blanca toca, su faz es dulce,es bella sin cesar, su faz es hermosa como un jaguar, su faz es divina como la roca.
Casta su faz, bajo la blanca toca, ennoblece este cantar popular, casta su faz,bajo la blanca tocar, ennoblece este cantar de boca.
Casta su faz, bajo la blanca toca, no dejaba errar;casta su mirar, no dejaba turbar;casta su hablar, no dejaba hacer;casta su boca.

Nada más. De aquel sueño, que pasó como un ala,
voló el agua como un ave frugalmente,
De aquel sueño,que pasó como una bala,
voló el viento como un átomo ardiente.

Nada más. De aquel sueño, que pasó como un ala,
no me devolviera la pálida piel caliente,
De aquel sueño,que pasó como una bala,
no me devolviera la herida en la frente.

Nada más. De aquel sueño, que pasó como un ala,
arranco todo lo que había en mi mente,
todo lo que me atormentaba y no cala,
lo que callo en el interior de mi presente.

Nada más. De aquel sueño, que pasó como un ala,
salió un pensamiento que viniera al frente,
un pensamiento que pasó como una bala,
salió un rayo que se encendió en mi mente.

Nada más. De aquel sueño, que pasó como un ala,
como un beso que se derrumbó de mi mente,
y que se fue al pasado como una bala,
y desapareció del mapa como un ente.

Nada más. De aquel sueño, que pasó como un ala,
un ave se paró sola en un parapente,
De aquel sueño,que pasó como una bala,
un viento se detuvo en el mar de enfrente.

Es inútil que luche por quitarme del pecho,
Es muy difícil que yo huya de mis traiciones,
Es demasiado tarde para lo que he hecho,
Es demasiado tarde para mis decisiones.

Es inútil que luche por quitarme del pecho,
que niegue repetidamente mis opiniones,
que trague de nuevo mi entusiasmo deshecho,
que rechace de nuevo todas mis negaciones.

Es inútil que luche por quitarme del pecho,
que no sea la tiranía de las pasiones,
que se ría de mí de lo que yo le he hecho,
Es inútil que se enoje por mis canciones.

Es inútil que luche por quitarme del pecho,
es inútil que llore por tus provocaciones,
es inútil que te diga qué es lo más hecho,
es inútil que afirme mis acusaciones.

Es inútil que luche por quitarme del pecho,
que luche por alcanzarme con sus oraciones,
que luche por bajarme del caballo derecho,
que me meta en mi cama con sus peticiones.

Es inútil que luche por quitarme del pecho,
que me refugie en mi casa de ilusiones,
que me grite a voces que quiero y no hecho,
que me arregle los días sin palpitaciones.

Del Monserrat en la penumbra undosa, Del Monserrat en la luz crepuscular, Del Monserrat en la vida de Aznar, Del Monserrat en la noche ansiosa.
Del Monserrat en la penumbra undosa, Del Monserrat en la niebla uncular, Del Monserrat en la luna anular, Del Monserrat en la noche brumosa.
Del Monserrat en la penumbra undosa, Del Monserrat en la noche un lugar, Del Monserrat en la luz un despertar, Del Monserrat en el sol una rosa.
Del Monserrat en la penumbra undosa, Del Monserrat con los dedos sin borrar, Del Monserrat con las ganas de cantar, Del Monserrat de la roca filosa.
Del Monserrat en la penumbra undosa, Del Monserrat silente en el Solar, Del Monserrat dolido en el remar, Del Monserrat cautivo en la prosa.
Del Monserrat en la penumbra undosa, Del Monserrat cuando hay que claudicar, Del Monserrat hasta que hay que llorar, Del Monserrat con mantilla larosa.

Table 8: Full list of six candidate poems for Spanish generated by our system for the five lines chosen at random in the Spanish dataset. No cherry-picking was done on this table. From these poems, we did select one for each line manually, as shown in Table 6 in the main body of the paper.

Gu biok lagun handiak gara, anaia,aita,semea, eta bion ideologia, gure identitatea, konpartitzen dugu.Batzuetan, zaila da bat esatea, besteak ulertzea,benetan, zein ahula den bestea.
Gu biok lagun handiak gara, bata zinemazalea, bestea musikaria bera, biek dugu lanbidea, Denok egiten dugu aurrera, dena da inportantea, dena dago egiteko era, izaten da idazlea.
Gu biok lagun handiak gara, argazkilari zalea, eta asko gustatzen zait bera, nire bikotekidea, argazki kamera eskuetan, ezaguna izatea, gogoratzen dudan argazkia, lagunekin ikustea.
Gu biok lagun handiak gara, Esne Beltzako kidea, anai zaharra eta arreba, Jon eta nire semea, eta bi panpin gaude honetan, Oihan Vega da bestea, Hesian taldeko kantaria, Sorotan Bele taldea.
Gu biok lagun handiak gara, katuak ardi taldea, tomateak eta haragia, soja eta laborea, oilaskoak fruitua lortzeko, lekalearen jeztea, baratzea animalietan, pinpilinpauxa zaintzea.
Gu biok lagun handiak gara, baina gure bikotea, itxura guztien arabera, ez da bideko jendea, Bidaia ez da gauza berria, ez da hala esatea, Baina batzuetan,batzuetan, izan daiteke jabea.

Nahiz kulunpio pila bat egon, galtza belauna behera, gero eskailerak,hortz parea, sokak,besoak atzera, kautxuzko motxilak bizkotxetan, eta astindu ederra, bularra,pultsuko aurpegia, izterra eta izterra.
Nahiz kulunpio pila bat egon, ez da oso gauza bera, autoa hartu eta kotxea, leku batetik bestera, beste leku batera sartzeko, jendearekin batera, egiten dugun bide horretan, kotxeak izorratzera.
Nahiz kulunpio pila bat egon, aspiranteen antzera, gorputza sentsuala da ia, betirako izaera, eta erretzen duzu airea, ez zara esterilera, kaka egiten duzu oinetan, likidoa da gainera.
Nahiz kulunpio pila bat egon, eguzkiak sikiera, aukera du ondo goxatzeko, eta ez beti gainera, baita asteko egun denetan, baita hemendik aurrera, ilargi erdiko orduetan, eta hori da ederra.
Nahiz kulunpio pila bat egon, komunera lasaitzera, komunera salto egiteko, irakurtzera,jotzera, txorimaloetaz gozatzea, konpainia egitera, burrunba burrunbatsuaz janztea, koadrilan afaltzera.
Nahiz kulunpio pila bat egon, bada zerbaiten plazera, dena zure eskura uztea, dakizuna arabera, izan zaitez supergizakia, izan nahi izatera, hau da,zuhaur,dena emateko, munduarekin batera.

Table 9: Full list of six candidate poems for Basque generated by our system for the two lines chosen at random in the Basque dataset. No cherry-picking was done on this table. From these poems, we did select one for each line manually, as shown in Table 7 in the main body of the paper.

CombLM: Adapting Black-Box Language Models through Small Fine-Tuned Models

Aitor Ormazabal[1] **Mikel Artetxe**[2] **Eneko Agirre**[1]
[1]HiTZ Center, University of the Basque Country (UPV/EHU) [2]Reka AI
aitor.ormazabal@ehu.eus mikel@reka.ai e.agirre@ehu.eus

Abstract

Methods for adapting language models (LMs) to new tasks and domains have traditionally assumed white-box access to the model, and work by modifying its parameters. However, this is incompatible with a recent trend in the field, where the highest quality models are only available as black-boxes through inference APIs. Even when the model weights are available, the computational cost of fine-tuning large LMs can be prohibitive for most practitioners. In this work, we present a lightweight method for adapting large LMs to new domains and tasks, assuming no access to their weights or intermediate activations. Our approach fine-tunes a small white-box LM and combines it with the large black-box LM at the probability level through a small network, learned on a small validation set. We validate our approach by adapting a large LM (OPT-30B) to several domains and a downstream task (machine translation), observing improved performance in all cases, of up to 9%, while using a domain expert 23x smaller.

1 Introduction

Natural language processing (NLP) has witnessed remarkable progress in recent years thanks to the development of increasingly powerful LMs (Brown et al., 2020; Andrew and Gao, 2007; Chowdhery et al., 2022; Touvron et al., 2023). Since these models are usually generalists, it is often of interest to adapt them to new domains, underrepresented or not found in the original training data. Typically, domain adaptation techniques assume white-box access to the model parameters, for example by fine-tuning on a particular target domain (Gururangan et al., 2020).

However, this approach has become increasingly infeasible given the ongoing paradigm shift in the field—state-of-the-art models like GPT-4 and PaLM-2 are only accessible as black-boxes through inference APIs and, even when the model weights are available, the computational cost of fine-tuning large models can be prohibitive. Consequently, domain adaptation methods that cannot leverage the power of black-box LLMs are likely to fall behind.

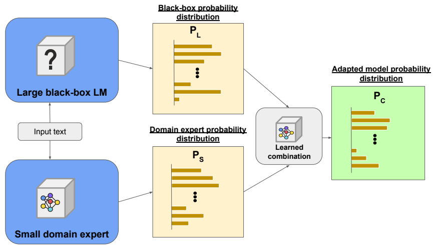

Black-box probability distribution
P_L
Large black-box LM
Input text
Small domain expert
Domain expert probability distribution
P_S
Learned combination
Adapted model probability distribution
P_C

Figure 1: **Illustration of our approach.** We leverage a large black-box LM and a small white-box LM, fine-tuned on a domain-specific corpus. We combine both models' outputs at the probability level, through a combination function learned on a small fitting set, requiring very little compute. The resulting model adapts the large black-box to the target domain, performing better than either of the original ones.

In this work, we propose a simple and lightweight approach to adapt black-box LMs to new domains, without requiring access to weights or intermediate activations. Our method consists of two main steps: (1) training a small, white-box model on the desired target domain, and (2) learning a function that combines the probability distributions from the large black-box LM and the small domain expert LM, producing a new probability distribution. The combination function is a small neural network that is trained on a small validation dataset.

We evaluate our method by adapting a black-box model to three distinct domains and a downstream task—machine translation (MT). In all cases, we observe that the combined model outperforms both the large black-box model and the small domain expert. This shows that it is possible to adapt black-box LMs to new domains, opening an exciting line of research.

2 Proposed method

Our approach works in two steps: (1) we train a small domain expert LM, and (2) we learn a function that combines the outputs of the domain expert LM and a large black-box LM at the probability level.

More concretely, an LM defines a probability distribution over the possible continuations of any given text. That is, given a sequence of tokens $\mathbf{x} = (x_1, x_2, ..., x_n) \in V^*$, where V is the model vocabulary, an LM parametrizes $P_{LM}(y_{next}|\mathbf{x})$, the probability that y_{next} is the continuation of $\mathbf{x}$ in a text. We let P_S denote our small domain expert LM, and P_L denote the large black-box generalist LM. Our combination function $\mathbf{f}$ defines a new combined probability distribution P_C: $P_C(y_{next}|\mathbf{x}) = \mathbf{f}(P_S(\cdot|\mathbf{x}), P_L(\cdot|\mathbf{x}))_{y_{next}}$. Here $\mathbf{f} : \mathbb{R}^{|V|} \times \mathbb{R}^{|V|} \rightarrow \mathbb{R}^{|V|}$ is a vector-valued function that receives full probability distributions, and outputs a new probability distribution.

To train the domain expert LM, we fine-tune a pre-trained model on a small domain-specific dataset. For the combination function, we consider several alternatives of varying capacity:

1. **Mean.** The arithmetic mean of the two distributions: $\mathbf{f}(\mathbf{y_1}, \mathbf{y_2}) = (\mathbf{y_1} + \mathbf{y_2})/2$.

2. **Constant-scalar**. A linear combination of the two input distributions, with a constant combination factor λ: $\mathbf{f}(\mathbf{y_1}, \mathbf{y_2}) = \lambda\mathbf{y_1} + (1 - \lambda)\mathbf{y_2}$.

3. **Constant-vector**. A token-wise version of the previous combination, where $\boldsymbol{\lambda} \in \mathbb{R}^{|V|}$ is a constant vector, and the combination factor varies per-token: $\mathbf{f}(\mathbf{y_1}, \mathbf{y_2}) \propto \boldsymbol{\lambda} \circ \mathbf{y_1} + (\mathbf{1} - \boldsymbol{\lambda}) \circ \mathbf{y_2}$, where $\circ$ is the Hadamard (elementwise) product. Note the proportionality instead of equality in the definition, as a re-normalization is required when combining distributions per-token.

4. **Entropy-scalar**. A scalar λ is predicted from the entropies of each distribution, $\lambda = g(\mathrm{H}(\mathbf{y_1}), \mathrm{H}(\mathbf{y_2}))$, and the output is a linear combination as in *constant-scalar*: $\mathbf{f}(\mathbf{y_1}, \mathbf{y_2}) = \lambda\mathbf{y_1} + (1 - \lambda)\mathbf{y_2}$. The function g is parametrized by a small neural network.

5. **Entropy-vector**. An token-wise version of the previous combination, where a vector $\boldsymbol{\lambda} = \mathbf{g}(\mathrm{H}(\mathbf{y_1}), \mathrm{H}(\mathbf{y_2})) \in \mathbb{R}^{|V|}$ is predicted , and then the per-token combination is done as in *constant-vector*.

6. **Full-scalar**. A single λ is predicted from full input distributions, $\lambda = g(\mathbf{y_1}, \mathbf{y_2})$, and then the output is a linear combination as in the constant combination: $\mathbf{f}(\mathbf{y_1}, \mathbf{y_2}) = \lambda\mathbf{y_1} + (1 - \lambda)\mathbf{y_2}$. The function g is parametrized by a small neural network.

7. **Full-vector**. Token-wise version of the previous combination, where a vector $\boldsymbol{\lambda} = \mathbf{g}(\mathbf{y_1}, \mathbf{y_2}) \in \mathbb{R}^{|V|}$ is predicted , and the per-token combination is done as in *constant-vector*.

On one end of the spectrum, the *mean* and *constant-scalar* combinations have very low capacity, having zero and one learnable parameters, respectively. On the other end, the *full* combinations can represent rich combination functions, taking advantage of the information in the full output distributions. The *entropy* combinations are motivated by the fact that we expect output distribution entropies to be informative to the combination function; intuitively, knowing how certain each model is should be helpful when deciding which model to give more weight to. Additionally, token-wise versions of each method further increase the capacity of the combination function. This setup allows us to study how important combination function capacity is for the performance of the adapted model, as well as how this relates to the amount of data used for learning the combination.

These combination functions can be learned without any access to the LMs' weights or internal states, and require only a forward pass through the small set used to train the combination network. We refer to the process of training the small network that parametrizes the combination function as fitting the combination function. Once the combination function is fit, the combined model outputs valid probability distributions over continuations, and can be used as a regular LM.

3 Experimental setup

3.1 Models

We use OPT-30B and OPT-1.3B (Zhang et al., 2022) as our large black-box and small white-box LMs, respectively. Our choice of OPT is motivated by the following reasons:

1. Both the small and large models must share the tokenizer in our current formulation.[1] Since we want to train the small domain experts by fine-tuning an existing model, we need a model family that has both large and small models sharing the same tokenizer, which OPT provides.

2. To rigorously determine what constitutes a new domain for the models, we need to know what data they were trained on, which is not public for most proprietary models behind APIs.[2]

We report results for the large model and the small fine-tuned model, which can be taken as the **baselines**, as well as their combination through our proposed method. For the parametrization of the combination functions, we use small neural networks, with the following architectures:

- **Constant-scalar:** A single neuron with no input, passed through a sigmoid to force it into $(0, 1)$.

- **Constant-vector:** A vector of neurons with no input, passed through a sigmoid to force it into $(0, 1)^{|V|}$.

- **Entropy-scalar:** Input layer is two-dimensional, consisting of both entropies, followed by 1D BatchNorm, two hidden layers of dimension 512, with ReLU non-linearities, and a one-dimensional output layer with a sigmoid non-linearity, to force it into $(0, 1)$.

- **Entropy-vector:** Input layer is same as for *entropy-scalar*, followed by 1D BatchNorm, two hidden layers of dimension 512, with ReLU non-linearities, and a $|V|$-dimensional output layer with a sigmoid non-linearity, to force it into $(0, 1)^{|V|}$.

- **Full-scalar:** Input layer is $2|V|$-dimensional, consisting on the concatenated output distributions for each model, followed by 1D BatchNorm, two hidden layers of dimension 512, with ReLU non-linearities, and a one-dimensional output layer with a sigmoid non-linearity, to force it into $(0, 1)$.

- **Full-vector:** Input layer same as for *full-scalar*, $2|V|$-dimensional, followed by 1D BatchNorm, two hidden layers of dimension 512, with ReLU non-linearities, and a $|V|$-dimensional output layer with a sigmoid non-linearity, to force it into $(0, 1)^{|V|}$.

We train all combination networks using the Adam optimizer and a learning rate of $2\mathrm{e}{-3}$ with the exception of *constant-vector*, for which we use a learning rate of $1\mathrm{e}{-2}$, and a batch size of 1024. We run optimization for a single epoch in all cases, as we found this to be enough in preliminary experiments.

Note that the **mean** combination function has no learnable parameters. Finally, we also report **max-prob oracle** results as the upper-bound, which simulates a perfect combination function that gives 100% of the weight to the best model for any given token.

3.2 Evaluation

For evaluation, we adapt our model for three new domains and a downstream task. The three new **domains** are defined by three datasets:

- The **Amazon Reviews** dataset (McAuley et al., 2015; He and McAuley, 2016), consisting of a large collection of reviews and ratings entered by users on the Amazon website.

- The **Enron Emails** dataset (Klimt and Yang, 2004), consisting of internal emails made public by the Federal Energy Regulatory Commission during the investigation of the Enron company.

- The **FreeLaw** subset of The Pile (Gao et al., 2021), consisting of a large collection of court opinions from federal and state courts.

For each dataset, we extract two sets of 1000 1024-token sequences, which we call *train-fit* and *test*, respectively, and use the rest for the train set. The *train-fit* sets are used to fit the combination functions, and we report perplexity on the *test* sets for evaluation. We use the train set to fine-tune OPT-1.3B using the Adam optimizer, a 1024-token sequence length, a fixed learning rate of $4\mathrm{e}{-4}$, and a batch size of $1024 * 90 = 92160$ tokens. In the

[1] Although it is possible to either adapt LMs to a new vocabulary or extend our approach to work with different tokenizers, that would add a new dimension to our experiments, separate from the core research question that we want to study.

[2] While this is not a problem for applying our method in practice, it does rule out proprietary black-box models for scientific study.

	Amazon	Enron	Freelaw
OPT-1.3B FT	17.00	3.30	4.98
OPT-30B	20.37	5.53	6.50
Mean	15.88	3.47	4.92
Constant-scalar	15.80	3.27	4.84
Constant-vector	15.62	3.31	4.82
Entropy-scalar	15.50	**3.24**	4.78
Entropy-vector	15.41	3.24	**4.76**
Full-scalar	**15.36**	3.27	4.79
Full-vector	15.43	3.27	4.79
Max-prob (oracle)	12.59	2.89	4.12

Table 1: **Domain adaptation results (perplexity).** By combining a small domain expert and large general model, we achieve better perplexities than either of the original models.

	en-de	en-cs	de-en	cs-en	avg
OPT-1.3B FT	52.36	32.66	67.95	60.47	53.36
OPT-30B	54.77	29.21	68.45	61.83	53.56
Mean	57.62	35.34	**69.84**	63.62	56.61
Constant-scalar	57.73	35.08	69.70	63.70	56.56
Constant-vector	57.71	34.69	69.60	63.64	56.41
Entropy-scalar	57.87	35.18	69.59	63.88	56.63
Entropy-vector	**58.11**	**35.41**	69.44	**64.06**	**56.76**
Full-scalar	57.98	35.06	69.57	63.59	56.55
Full-vectors	58.02	35.31	69.66	63.37	56.59

Table 2: **MT results (BLEURT).** The learned combinations significantly outperforms both models in a downstream task, often by a substantial margin.

case of Enron Emails we fine-tuned for a single epoch, corresponding to 3000k steps. For Amazon Reviews and FreeLaw we performed 30k steps, and had to stop well before reaching the first epoch, due to compute constraints. Unless otherwise stated, the full *train-fit* sets are used to fit the combination functions.

For **downstream evaluation**, we experiment on English-Czech and English-German MT using the WMT21 dataset (Barrault et al., 2020). We create a training set by verbalizing all the sentence pairs and concatenating them into a single corpus. Details of the verbalization templates can be found in Appendix A. We create a validation set following the same procedure on the WMT20 test set (Akhbardeh et al., 2021), and extract a *train-fit* set of 1000 1024-token sequences for fitting the combination functions, as we do in domain adaptation. Following the recommended practice in the area (Freitag et al., 2022), we use BLEURT (Sellam et al., 2020) on the WMT21 test set as our evaluation metric, and report additional results with BLEU (Papineni et al., 2002) in Appendix B. We used 3-shot prompting for evaluation, as longer sequence lenghts resulted in OOM issues in our hardware. We use the training set to fine-tune OPT-1.3B using the exact same settings described above. We train for 2k steps, corresponding to a total of around 2.5 million parallel sentences.[3]

4 Results

We next present our main results on domain adaptation (§4.1) and MT (§4.2).

4.1 Domain adaptation

We report domain adaptation results in Table 1. We observe that the combined models are able to achieve substantially lower perplexities than either of the individual models. Even simple averaging works remarkably well, improving over both baselines in Amazon Reviews and FreeLaw, but learned combinations perform best. The *entropy-scalar* combination works best across the board, achieving a relative improvement in perplexity of 9% in Amazon Reviews, 2% in Enron Emails and 4% in FreeLaw over the best single model. This supports our hypothesis that output distribution entropies are informative to the combination function. However, higher capacity combination functions like *full-scalar* work better in some cases, as is the case for Amazon Reviews.

Overall, our results show that the adapted model is able to leverage domain-specific knowledge in the small model, as well as the knowledge in the large generalist model, in order to improve over either of them. However, there is still a significant gap between the adapted models and the max-prob oracle, suggesting gains could still be made through a better combination function.

4.2 Machine translation

Table 2 reports downstream results on MT. As for domain adaptation, all the learned combinations outperform both the small fine-tuned model and the large black-box model. This shows that our approach can work for adaptation to downstream tasks, and is not limited to domain adaptation. Once again, the simple *mean* combination per-

[3] Although the full combined training set for English-German and English-Czech is bigger than 2.5M parallel sentences, we were interested in simulating the setting where limited translation data is available. Given enough parallel data, one can train a strong translation system from scratch, without having to adapt a generalist model.

forms very well, obtaining the second best results after *entropy-vector*. In any case, the combination function has a relatively small impact in MT, and even the worst performing approach brings large improvements over the baseline.

5 Analysis

In this section, we study the following aspects of our approach:

- How dependent is the quality of the resulting model on the amount of data used to fit the combination function?
- How dependent is the quality of the resulting model on the amount of data used to fine-tune the small LM?
- How much is general language modeling performance degraded by domain adaptation?
- Is the learned combination interpretable?

5.1 Effect of the amount of data for fitting

In order to study how the performance of the adapted model varies with respect to the amount of data used to fit the combination function, we fit each combination function three times, on a varying number of tokens. We report results for the Amazon Reviews dataset in Table 3, and additional results in Appendix B.

As expected, performance improves with more training data. However, the difference varies across methods. For example, *constant-scalar*, which has a very low capacity, performs equally well when trained on 100 or 1000 sequences. On the other hand, the *full-scalar* and *full-vector* functions, that take the entire probability distribution as input, benefit from more training sequences. The *entropy-scalar* combination strikes a good balance, performing well across the board, and retaining strong performance when fit on as little as 100 sequences.

5.2 Effect of fine-tuning steps

Figure 2 shows the performance of the adapted models, when fine-tuning the small model for a varying number of sequences. At step 0 (i.e., before fine-tuning begins), the small LM corresponds to vanilla OPT-1.3B, which performs considerably worse than OPT-30B on Amazon Reviews. Even in that case, *entropy-scalar* performs on par with OPT-30B, while *mean* is slightly worse. This shows that learnable combination functions are able to avoid any loss in performance when combining with a poor domain expert. At the same time, it is also remarkable that the combination of vanilla OPT-1.3B and OPT-30B is not better than OPT-30B alone. This can also be seen in Table 4, which compares using vanilla OPT-1.3B and fine-tuned OPT-1.3B as the small model. This shows that our reported improvements do not solely come from an ensembling effect, and our proposed approach effectively combines the power of the large LM and the domain expertise of the small LM.

	100	**500**	**1000**
OPT-1.3B FT	17.00	17.00	17.00
OPT-30B	20.37	20.37	20.37
Mean	15.88	15.88	15.88
Constant-scalar	15.80	15.80	15.80
Constant-vector	15.80	15.66	15.62
Entropy-scalar	15.51	15.50	15.50
Entropy-vector	15.52	15.45	15.41
Full-scalar	15.63	15.40	15.36
Full-vector	15.71	15.49	15.43

Table 3: **Perplexity on Amazon Reviews**, using a different number of sequences to fit the combination function. Perplexity improves with the number of sequences, but results are already strong with only 100 sequences.

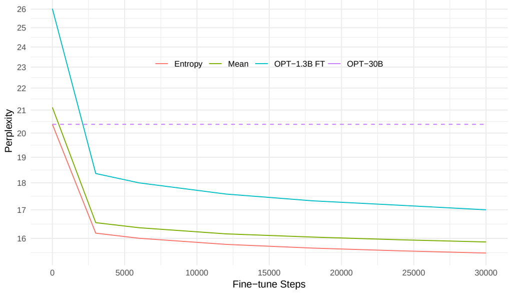


Figure 2: **Perplexity on Amazon Reviews**, varying the amount of fine-tuning steps.

In addition, we observe that our combined LM substantially improves upon each individual LM as early as step 3000. In fact, the gap between the small fine-tuned LM and our combined LM slightly narrows as training progresses. For instance, for *entropy-scalar*, the gap between the small LM and the combined LM is 2.18 perplexity points at step 3000 (12% relative improvement), which goes down to 1.5 for the fully fine-tuned model (9% relative improvement). This is intuitive, as the more data is available in the target domain, the less useful will be integrating the general knowledge in the large LM.

	Orig	FT
OPT-1.3B	26.03	17.00
OPT-30B	20.37	20.37
Mean	21.12	15.88
Constant-scalar	20.28	15.80
Constant-vector	20.55	15.62
Entropy-scalar	20.37	15.51
Entropy-vector	20.30	15.44
Full-scalar	20.26	15.41
Full-vector	20.30	15.48

Table 4: **Perplexity on Amazon Reviews**, using either original OPT-1.3B or fine-tuned OPT-1.3B as the small LM. The combination methods barely improve upon OPT-30B when using the former, showing that our approach does not only work due to an ensembling effect.

	Amazon-fit		**Mixin-fit**	
	Amazon	**Pile**	**Amazon**	**Pile**
OPT-1.3B FT	17.00	19.78	17.00	19.78
OPT-30B	20.37	6.82	20.37	6.82
Mean	15.88	7.72	15.88	7.72
Constant-scalar	15.80	8.35	16.52	7.08
Constant-vector	15.62	8.38	15.89	7.18
Entropy-scalar	15.50	7.35	15.80	6.94
Entropy-vector	15.41	8.53	15.61	6.92
Full-scalar	15.36	9.31	15.45	6.85
Full-vector	15.43	10.07	15.48	6.91

Table 5: **Perplexity on Amazon Reviews and The Pile**, using either the former to fit the combination function (amazon-fit), or the concatenation of the two (mixin-fit).

5.3 Effect on general language modeling

We are also interested in measuring how well the adapted models retain the general language modeling ability of the original large model. We use perplexity on **The Pile** (Gao et al., 2021) as a proxy of general language modeling performance, as it is a large collection of many datasets from different domains, often used to train generalist LMs (Black et al., 2022; Biderman et al., 2023). To this end, we also extract random *train-fit* and *test* subsets from The Pile. While some subsets of The Pile are also present in the training data for OPT, we do not measure performance on The Pile as a benchmark for model quality, and are only interested in it as a proxy for degradation in general language modeling ability of the adapted models.

We compare fitting the combination function on the target domain *train-fit*, as done throughout the paper, as well as on the combination of the target domain and The Pile *train-fit* sets. Table 5 reports results for Amazon Reviews, and full results can be found in Appendix B.

When fitting the combination function on Amazon Reviews, we observe a significant degradation on The Pile. However, different combination methods behave differently in this regard. For example, *entropy-scalar* and *full-vector* perform similarly in Amazon Reviews (15.50 vs 15.43), but the former performs much better on The Pile (7.35 vs 10.07). It is also remarkable that The Pile perplexity of the combined model remains far better than the small fine-tuned LM (e.g. 7.35 for *entropy-scalar* vs 19.78 for the small LM), while also performing better in-domain.

When fitting the combination function on the mixin set, we observe that performance on The Pile is almost entirely preserved, at the expense of a slight degradation on Amazon Reviews. For example, for *full-scalar*, the combination fit on the mixin set achieves a perplexity of 15.45 on Amazon Reviews and 6.85 on The Pile, both within 0.1 of the best results for each dataset.

Overall, these results show that a large model can be adapted to a particular domain while mitigating degradation in the general domain by mixing in-domain and general text to train the combination function. Additionally, we find that different combination methods exhibit different behavior when it comes to general performance degradation, even when they exhibit similar in-domain performance.

5.4 Is the model combination interpretable?

We next analyze whether the weights given to each model by the combination function are interpretable. Figure 3 illustrates this over a random sample from Amazon Reviews: we show which tokens are better predicted by each model, along with which model is given a higher weight for each token. Although we do not identify a clear pattern for which tokens are better predicted by each model, we do observe that the coloring in the top and the bottom visualizations match quite closely. This means that the learned combination function is quite good at predicting when each model should be given a higher weight.[4]

In order to quantitatively analyze this, we measure the Spearman correlation between the weight given by the combination function, and the actual difference in log probabilities for each token. Re-

[4] A perfect combination function (corresponding to the max-prob oracle in Table 1) would always give 100% of the weight to the best model for any given token, and both images would match up perfectly.

		Domain	Pile
Amazon	**Entropy-scalar**	0.59	0.71
	Full-scalar	0.44	0.32
Freelaw	**Entropy-scalar**	0.49	0.75
	Full-scalar	0.33	0.32
Enron	**Entropy-scalar**	0.54	0.75
	Full-scalar	0.25	0.30

Table 6: **Spearman correlation between the log-probability difference of the LMs and the weight given by combination function.** Larger values mean that the learned combination is closer to the ideal oracle weighting. Rows represent adapted models on different domains and combination functions, fit on the in-domain *train-fit*.

sults are shown in Table 6. We limit our analysis to *entropy-scalar* and *full-scalar*, as they are the only ones that output a single combination factor that depends on the input. We observe significant correlations for all datasets, with *entropy-scalar* achieving better correlation than *full-scalar*, especially on The Pile. This is consistent with the results in Table 5, where *full-scalar* suffers a bigger performance loss on The Pile. Somewhat surprisingly, correlation for *entropy-scalar* is better on The Pile than on the in-domain dataset, even though the combination function is fit on the in-domain *train-fit*. One possible explanation is that The Pile better represents the training distribution of the large LM, making it better calibrated on it, which makes it easier for *entropy-scalar* to make predictions.

6 Related work

We present related work on domain adaptation of LMs (§6.1), and language modeling through domain experts (§6.2).

6.1 Domain adaptation of LMs

Domain adaptation of LMs is an extensively studied line of research. Traditional approaches include fine-tuning the model on domain-specific corpora, (Devlin et al., 2019; Liu et al., 2019; Gururangan et al., 2020), data selection on the original general corpus (Aharoni and Goldberg, 2020; van der Wees et al., 2017), and adapting or extending the tokenizer to achieve better performance on the target domain (Sachidananda et al., 2021).

Although effective, these full fine-tuning techniques are often infeasible at scale due to the excessive compute required. Some approaches aim to reduce the resources required to fine-tune large models through parameter-efficient adaptation techniques, such as adapters (Houlsby et al., 2019), soft-prompt tuning (Liu et al., 2022), or low-rank adaptation (Hu et al., 2022). However, all of these techniques require white-box access to the original model and full backward passes, making them incompatible with black-box models.

In contrast, discrete prompt tuning approaches allow for treating the large model as a black-box (Shin et al., 2020; Sun et al., 2022; Zhang et al., 2023; Cheng et al., 2023). However, these approaches have only been proven in the limited setting of retrieving zero- or few-shot prompts that improve performance in a set of NLP tasks that the base black-box is already capable of performing, as opposed to a general method of black-box model adaptation.

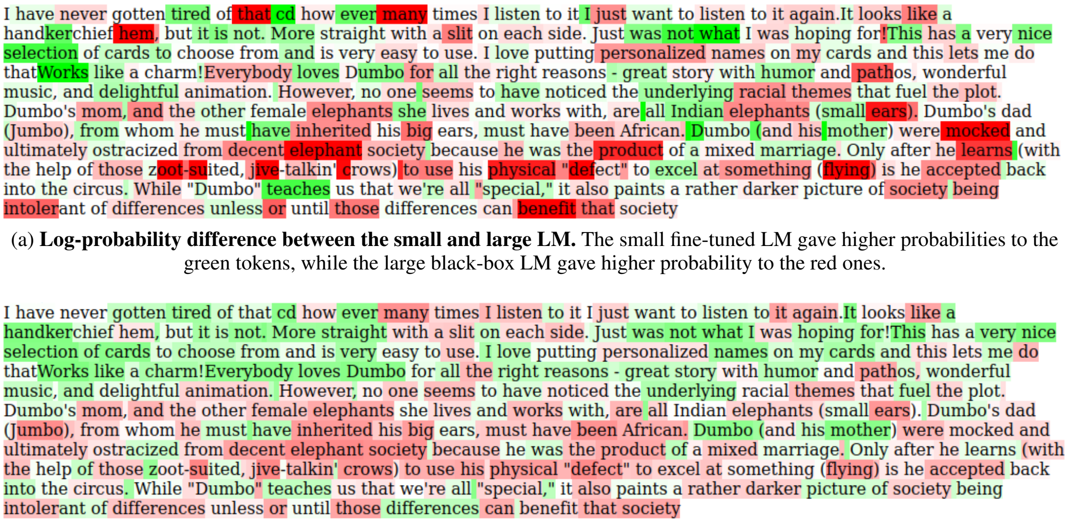
Concurrent to our work, Huang et al. (2023) propose leveraging KNN retrieval from a data-store to augment an existing black-box LM. However, they only experiment with small GPT2 models as the black-box, and the adaptation depends on finding an adequate datastore, limiting application to downstream tasks such as MT.

6.2 Domain experts for language modeling

Another line of research explores language modeling through a combination of separate domain experts. Li et al. (2022) achieve better performance than compute-matched single transformer models and highly parallel pre-training, by training independent domain experts, and combining them at the parameter level at inference time. Gururangan et al. (2023) extend this approach to automatically discovered domain clusters. Other approaches replace components of the transformer network with independent domain-dependent modules, as is the case of Gururangan et al. (2022) for metadata-defined domains, or Pfeiffer et al. (2022) for per-language modules. All of these are pre-training approaches and seek to train better or more efficient LMs, but cannot leverage existing powerful black-box models. Our work, in contrast, seeks to adapt an existing powerful black-box through leveraging a much smaller domain expert.

7 Conclusions

In this work, we present a method for adapting black-box LMs to new domains and tasks, requiring access to probability-level outputs only. We first fine-tune a small domain expert white-box LM



I have never gotten tired of that cd how ever many times I listen to it I just want to listen to it again.It looks like a handkerchief hem, but it is not. More straight with a slit on each side. Just was not what I was hoping for!This has a very nice selection of cards to choose from and is very easy to use. I love putting personalized names on my cards and this lets me do thatWorks like a charm!Everybody loves Dumbo for all the right reasons - great story with humor and pathos, wonderful music, and delightful animation. However, no one seems to have noticed the underlying racial themes that fuel the plot. Dumbo's mom, and the other female elephants she lives and works with, are all Indian elephants (small ears). Dumbo's dad (Jumbo), from whom he must have inherited his big ears, must have been African. Dumbo (and his mother) were mocked and ultimately ostracized from decent elephant society because he was the product of a mixed marriage. Only after he learns (with the help of those zoot-suited, jive-talkin' crows) to use his physical "defect" to excel at something (flying) is he accepted back into the circus. While "Dumbo" teaches us that we're all "special," it also paints a rather darker picture of society being intolerant of differences unless or until those differences can benefit that society

(a) **Log-probability difference between the small and large LM.** The small fine-tuned LM gave higher probabilities to the green tokens, while the large black-box LM gave higher probability to the red ones.

I have never gotten tired of that cd how ever many times I listen to it I just want to listen to it again.It looks like a handkerchief hem, but it is not. More straight with a slit on each side. Just was not what I was hoping for!This has a very nice selection of cards to choose from and is very easy to use. I love putting personalized names on my cards and this lets me do thatWorks like a charm!Everybody loves Dumbo for all the right reasons - great story with humor and pathos, wonderful music, and delightful animation. However, no one seems to have noticed the underlying racial themes that fuel the plot. Dumbo's mom, and the other female elephants she lives and works with, are all Indian elephants (small ears). Dumbo's dad (Jumbo), from whom he must have inherited his big ears, must have been African. Dumbo (and his mother) were mocked and ultimately ostracized from decent elephant society because he was the product of a mixed marriage. Only after he learns (with the help of those zoot-suited, jive-talkin' crows) to use his physical "defect" to excel at something (flying) is he accepted back into the circus. While "Dumbo" teaches us that we're all "special," it also paints a rather darker picture of society being intolerant of differences unless or until those differences can benefit that society

(b) **Weight given to each model by *entropy-scalar*.** Tokens for which a higher weight was assigned to the small fine-tuned LM are shown in green, while tokens for which the large black-box was given a higher weight are shown in red.

Figure 3: **Difference between the small fine-tuned LM and the large black-box LM according to log-probability (a) and predicted weight (b).** The closer the two match, the better the learned combination is at predicting which model will be "right" for a given token. The text sample was chosen randomly from the Amazon Reviews testset.

on a domain-specific corpus, and then combine it with the large black-box through a combination function learned on a small fitting set, yielding an adapted LM. Additionally, our method requires only access to probability level outputs, and thus allows to leverage powerful models optimized for inference or behind APIs, without the need for white-box access to the weights. We experiment on several datasets and a downstream task, as well as performing extensive analysis of our method, reaching several conclusions:

- By combining a small domain expert and a large black-box model, the combined model outperforms either of the original ones in all cases, by as much as 9% perplexity for domain adaptation, and 6% BLEURT for MT, showing the effectiveness of our approach.
- While higher capacity combination functions can perform better when more data is used to learn the combination, lower capacity combination methods remain competitive, and perform better when learned on little data. In particular, the entropy based combinations, *entropy-scalar* and *entropy-vector*, perform well across the board, even when fit on as little as 100 sequences.
- Our approach is effective even when little is data available to fine-tune the domain expert. In fact, the gains are biggest in this scenario, as the advantage of leveraging a good black-box generalist decreases when a big in-domain corpus is available.
- While adaptation to new domains incurs a loss of general language modeling ability, this varies per combination method, and seems to be largely mitigated by augmenting the small set on which the combination function is fit.

While our approach is effective, observed performance is still not close to the max prob oracle, which represents the ideal system where 100% of the weight is given to the best model at each time step. In future work, we would like to investigate the reasons for this gap, and potential ways of addressing it.

References

Roee Aharoni and Yoav Goldberg. 2020. Unsupervised domain clusters in pretrained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7747–7763, Online. Association for Computational Linguistics.

Farhad Akhbardeh, Arkady Arkhangorodsky, Magdalena Biesialska, Ondřej Bojar, Rajen Chatterjee, Vishrav Chaudhary, Marta R. Costa-jussa, Cristina España-Bonet, Angela Fan, Christian Federmann, Markus Freitag, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Leonie Harter, Kenneth Heafield, Christopher Homan, Matthias Huck, Kwabena Amponsah-Kaakyire, Jungo Kasai, Daniel Khashabi, Kevin Knight, Tom Kocmi, Philipp Koehn, Nicholas Lourie, Christof Monz, Makoto Morishita, Masaaki Nagata, Ajay Nagesh, Toshiaki Nakazawa, Matteo Negri, Santanu Pal, Allahsera Auguste Tapo, Marco Turchi, Valentin Vydrin, and Marcos Zampieri. 2021. Findings of the 2021 conference on machine translation (WMT21). In *Proceedings of the Sixth Conference on Machine Translation*, pages 1–88, Online. Association for Computational Linguistics.

Galen Andrew and Jianfeng Gao. 2007. Scalable training of L_1-regularized log-linear models. In *Proceedings of the 24th International Conference on Machine Learning*, pages 33–40.

Loïc Barrault, Magdalena Biesialska, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Matthias Huck, Eric Joanis, Tom Kocmi, Philipp Koehn, Chi-kiu Lo, Nikola Ljubešić, Christof Monz, Makoto Morishita, Masaaki Nagata, Toshiaki Nakazawa, Santanu Pal, Matt Post, and Marcos Zampieri. 2020. Findings of the 2020 conference on machine translation (WMT20). In *Proceedings of the Fifth Conference on Machine Translation*, pages 1–55, Online. Association for Computational Linguistics.

Stella Biderman, Hailey Schoelkopf, Quentin Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal. 2023. Pythia: A suite for analyzing large language models across training and scaling.

Sid Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, Michael Pieler, USVSN Sai Prashanth, Shivanshu Purohit, Laria Reynolds, Jonathan Tow, Ben Wang, and Samuel Weinbach. 2022. Gpt-neox-20b: An open-source autoregressive language model.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Daixuan Cheng, Shaohan Huang, Junyu Bi, Yuefeng Zhan, Jianfeng Liu, Yujing Wang, Hao Sun, Furu Wei, Denvy Deng, and Qi Zhang. 2023. Uprise: Universal prompt retrieval for improving zero-shot evaluation.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. Palm: Scaling language modeling with pathways.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Markus Freitag, Ricardo Rei, Nitika Mathur, Chi-kiu Lo, Craig Stewart, Eleftherios Avramidis, Tom Kocmi, George Foster, Alon Lavie, and André F. T. Martins. 2022. Results of WMT22 metrics shared task: Stop using BLEU – neural metrics are better and more robust. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 46–68, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. 2021. The pile: An 800gb dataset of diverse text for language modeling. *CoRR*, abs/2101.00027.

Suchin Gururangan, Mike Lewis, Ari Holtzman, Noah A. Smith, and Luke Zettlemoyer. 2022. DEMix layers: Disentangling domains for modular language modeling. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5557–5576, Seattle, United States. Association for Computational Linguistics.

Suchin Gururangan, Margaret Li, Mike Lewis, Weijia Shi, Tim Althoff, Noah A. Smith, and Luke Zettlemoyer. 2023. Scaling expert language models with unsupervised domain discovery.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Ruining He and Julian McAuley. 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *Proceedings of the 25th International Conference on World Wide Web*, WWW '16, page 507–517, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Yangsibo Huang, Daogao Liu, Zexuan Zhong, Weijia Shi, and Yin Tat Lee. 2023. $k$nn-adapter: Efficient domain adaptation for black-box language models.

Bryan Klimt and Yiming Yang. 2004. The enron corpus: A new dataset for email classification research. In *Machine Learning: ECML 2004*, pages 217–226, Berlin, Heidelberg. Springer Berlin Heidelberg.

Margaret Li, Suchin Gururangan, Tim Dettmers, Mike Lewis, Tim Althoff, Noah A. Smith, and Luke Zettlemoyer. 2022. Branch-train-merge: Embarrassingly parallel training of expert language models.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. 2022. P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 61–68, Dublin, Ireland. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton van den Hengel. 2015. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '15, page 43–52, New York, NY, USA. Association for Computing Machinery.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Jonas Pfeiffer, Naman Goyal, Xi Lin, Xian Li, James Cross, Sebastian Riedel, and Mikel Artetxe. 2022. Lifting the curse of multilinguality by pre-training modular transformers. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3479–3495, Seattle, United States. Association for Computational Linguistics.

Vin Sachidananda, Jason Kessler, and Yi-An Lai. 2021. Efficient domain adaptation of language models via adaptive tokenization. In *Proceedings of the Second Workshop on Simple and Efficient Natural Language Processing*, pages 155–165, Virtual. Association for Computational Linguistics.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4222–4235, Online. Association for Computational Linguistics.

Tianxiang Sun, Yunfan Shao, Hong Qian, Xuanjing Huang, and Xipeng Qiu. 2022. Black-box tuning for language-model-as-a-service.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models.

Marlies van der Wees, Arianna Bisazza, and Christof Monz. 2017. Dynamic data selection for neural machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1400–1410, Copenhagen, Denmark. Association for Computational Linguistics.

Biao Zhang, Barry Haddow, and Alexandra Birch. 2023. Prompting large language model for machine translation: A case study.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. Opt: Open pre-trained transformer language models.

A MT verbalizations

We verbalize the MT task by first adding a prompt describing the task, and then adding several translation examples. We chunk the translation examples in blocks of 5, that is, adding 5 translation examples per verbalization. We use two different task descriptiopns, shown in Table 7, and alternate evenly between both variations to create the verbalized training corpus. For inference, we use verbalization #1 and draw 3 random translation pairs from the WMT21 development set to construct a 3-shot prompt. We draw the random translation pairs once, and keep the 3-shot prompt fixed for all models.

B Full results

Full results for all combination methods, dataset sizes, and evaluation settings are shown in Table 8. Table 9 reports additional MT results using BLEU.

Verbalization #1	Verbalization #2
Translate the following sentences from $L1 to $L2: $L1: $S1 $L2: $T1 $L1: $S2 $L2: $T2 $L1: $S3 $L2: $T3 $L1: $S4 $L2: $T4 $L1: $S5 $L2: $T5	Given a sentence in $L1, translate it to $L2: $L1: $S1 $L2: $T1 $L1: $S2 $L2: $T2 $L1: $S3 $L2: $T3 $L1: $S4 $L2: $T4 $L1: $S5 $L2: $T5

Table 7: Both verbalizations used for MT. $L1 $L2 represent the source and target languages, and $Si and $Ti represent the source and target sentences for the *i*th pair in the verbalization.

COMBINATION FUNTION FIT ON TARGET DOMAIN *train-fit*

Dataset	**Amazon Reviews**						**Enron Emails**						**Freelaw**					
#Fit sequences	**100**		**500**		**1000**		**100**		**500**		**1000**		**100**		**500**		**1000**	
Eval domain	**Dom.**	**Pil.**	**Dom.**	**Pil.**	**Dom.**	**Pil.**	**Dom.**	**Pil.**	**Dom.**	**Pil.**	**Dom.**	**Pil.**	**Dom.**	**Pil.**	**Dom.**	**Pil.**	**Dom.**	**Pil.**
Mean	15.88	7.72	15.88	7.72	15.88	7.72	3.47	7.45	3.47	7.45	3.47	7.45	4.92	7.56	4.92	7.56	4.92	7.56
Constant-scalar	15.80	8.35	15.80	8.35	15.80	8.35	3.27	9.89	3.27	9.75	3.27	9.63	4.84	8.98	4.84	8.90	4.84	8.90
Constant-vector	15.80	7.82	15.66	8.12	15.62	8.38	3.42	7.59	3.34	8.03	3.31	8.39	4.90	7.68	4.84	8.05	4.82	8.37
Entropy-scalar	15.51	7.30	15.50	7.51	15.50	7.35	3.24	8.30	3.24	8.10	3.24	8.02	4.78	7.84	4.78	8.12	4.78	8.33
Entropy-vector	15.52	8.10	15.45	8.20	15.41	8.53	3.25	8.22	3.24	8.53	3.24	8.18	4.80	8.05	4.77	8.05	4.76	8.05
Full-scalar	15.63	7.97	15.40	9.28	15.36	9.31	3.32	8.22	3.27	10.11	3.27	9.86	4.82	8.13	4.79	9.48	4.79	9.45
Full-vector	15.71	7.90	15.49	9.62	15.43	10.07	3.34	8.11	3.27	10.54	3.27	9.82	4.85	7.90	4.80	9.30	4.79	9.27
OPT-1.3B FT	17.00	19.78	17.00	19.78	17.00	19.78	3.30	12.73	3.30	12.73	3.30	12.73	4.98	15.55	4.98	15.55	4.98	15.55
OPT-30B	20.37	6.82	20.37	6.82	20.37	6.82	5.53	6.82	5.53	6.82	5.53	6.82	6.50	6.82	6.50	6.82	6.50	6.82
Max-prob (oracle)	12.59	5.93	12.59	5.93	12.59	5.93	2.89	5.89	2.89	5.89	2.89	5.89	4.12	5.75	4.12	5.75	4.12	5.75

COMBINATION FUNTION FIT ON MIX OF IN DOMAIN AND THE PILE *train-fit*

Dataset	**Amazon Reviews**						**Enron Emails**						**Freelaw**					
#Fit sequences	**200**		**1000**		**2000**		**200**		**1000**		**2000**		**200**		**1000**		**2000**	
Eval domain	**Dom.**	**Pil.**	**Dom.**	**Pil.**	**Dom.**	**Pil.**	**Dom.**	**Pil.**	**Dom.**	**Pil.**	**Dom.**	**Pil.**	**Dom.**	**Pil.**	**Dom.**	**Pil.**	**Dom.**	**Pil.**
Mean	15.88	7.72	15.88	7.72	15.88	7.72	3.47	7.45	3.47	7.45	3.47	7.45	4.92	7.56	4.92	7.56	4.92	7.56
Constant-scalar	16.56	7.06	16.47	7.10	16.52	7.08	3.57	7.22	3.56	7.24	3.56	7.24	5.18	6.92	5.16	6.94	5.15	6.96
Constant-vector	15.85	7.53	15.85	7.29	15.89	7.18	3.45	7.41	3.44	7.32	3.45	7.27	4.93	7.37	4.95	7.14	4.96	7.04
Entropy-scalar	15.87	6.92	15.71	6.98	15.80	6.94	3.38	6.98	3.35	7.02	3.36	7.01	4.91	6.76	4.88	6.80	4.92	6.75
Entropy-vector	15.69	7.07	15.66	6.96	15.61	6.92	3.36	7.12	3.34	7.04	3.35	6.98	4.89	6.92	4.83	6.90	4.85	6.78
Full-scalar	15.55	6.91	15.42	6.90	15.45	6.85	3.47	7.10	3.47	7.06	3.45	6.99	4.93	6.78	4.90	6.72	4.85	6.77
Full-vector	15.63	7.16	15.53	6.98	15.48	6.91	3.41	7.37	3.42	7.17	3.44	7.05	4.92	7.05	4.89	6.90	4.87	6.80
OPT-1.3B FT	17.00	19.78	17.00	19.78	17.00	19.78	3.30	12.73	3.30	12.73	3.30	12.73	4.98	15.55	4.98	15.55	4.98	15.55
OPT-30B	20.37	6.82	20.37	6.82	20.37	6.82	5.53	6.82	5.53	6.82	5.53	6.82	6.50	6.82	6.50	6.82	6.50	6.82
Max-prob (oracle)	12.59	5.93	12.59	5.93	12.59	5.93	2.89	5.89	2.89	5.89	2.89	5.89	4.12	5.75	4.12	5.75	4.12	5.75

Table 8: Full results for all combination methods, when fit on different amount of tokens, and on different domains. Note that the #Fit sequences is doubled when fitting the combination function on a mix of in-domain and Pile data, since the same number of tokens is drawn from each.

	en-de		en-cs		de-en		cs-en		avg	
	BLEURT	**BLEU**	**BLEURT**	**BLEU**	**BLEURT**	**BLEU**	**BLEURT**	**BLEU**	**BLEURT**	**BLE**
Mean	57.62	14.39	35.34	5.76	69.84	26.72	63.62	21.57	56.61	17.1
Constant-scalar	57.73	13.76	35.08	5.66	69.70	26.68	63.75	21.32	56.56	16.8
Constant-vector	57.71	13.88	34.69	5.28	69.60	26.65	63.64	21.41	56.41	16.8
Entropy-scalar	57.87	13.76	35.18	5.60	69.59	26.30	63.88	21.31	56.63	16.7
Entropy-vector	58.11	14.25	35.41	5.64	69.44	26.73	64.06	21.47	56.76	17.0
Full-scalar	57.98	14.22	35.06	5.52	69.57	25.86	63.59	20.50	56.55	16.5
Full-vector	58.02	14.11	35.31	5.19	69.66	26.13	63.37	20.96	56.59	16.6
OPT-1.3B FT	52.36	15.05	32.66	5.48	67.95	25.27	60.47	19.13	53.36	16.2
OPT-30B	54.77	9.64	29.21	3.13	68.45	24.08	61.83	18.49	53.56	13.8

Table 9: Full BLEU and BLEURT results for MT.