Ingeniaritza Konputazionala eta
Sistema Adimentsuak Unibertsitate Masterra

Konputazio Zientziak eta Adimen Artifiziala Saila

Master Tesia

Informazio espaziala aztertzen eredu multimodaletan

Eneko Atxa Landa

Zuzendaritza

Gorka Azkune Galparsoro
Informatika fakultatea, UPV/EHU

Ander Salaberria Saizar
Informatika fakultatea, UPV/EHU

eman ta zabal zazu
Universidad del País Vasco
Euskal Herriko Unibertsitatea
INFORMATIKA FAKULTATEA
FACULTAD DE INFORMÁTICA
Konputazio Ingeniaritza eta Sistema Adimentsuak Unibertsitate Masterra
Máster Universitario en Ingeniería Computacional y Sistemas Inteligentes

eman ta zabal zazu

Universidad del País Vasco
Euskal Herriko Unibertsitatea

INFORMATIKA FAKULTATEA
FACULTAD DE INFORMÁTICA

Master Tesia

Konputazio Ingeniaritza eta Sistema Adimentsuak Unibertsitate Masterra

Informazio espaziala aztertzen eredu multimodaletan

Eneko Atxa Landa

Zuzendariak
Gorka Azkune Galparsoro
Ander Salaberria Saizar

2022.eko irailaren 5

Esker onak / Agradecimientos

Eskerrak eman nahi dizkiet nire guraso eta anaiari, gidatu baino, gidari izaten irakasteagatik.

Laburpena

Dokumentu honetan deskribatzen da Eneko Atxa Landa ikaslearen master amaierako lana osatu duen ikerketa prozesua. Irudiak eta testua prozesatzen dituzten transformer multimodalak aztertu dira, irudiak prozesatzeko garaian objektuen posizioa kodetzeko duten eran sakonduz.

Ikerketa hori egiteko, objektuen posizioa kodetzeko modu, edo *spatial embedding*, desberdinak konparatu dira elkarren artean. Oinarritzat VisualBERT izeneko transformer multimodal bat hartu da, zeinak ez duen *spatial embeddingik* erabiltzen izatez, eta hainbat *embedding* inplementatu dira eta elkarren artean konparatu.

Visual question answering (VQA) hartu da konparatzeko erabiliko den ataza bezala, zeinetan irudi bat eta honen gaineko galdera bat hartuta galderari erantzun behar zaion. Bertan ikusiko da ea *spatial embedding* desberdinek nolako eragina duten galderei erantzuterako garaian. VQA v2.0 datu-multzoa erabiliko da probak egiteko hasieran, atazari lotua dagoen datu-multzoa izanik. Ondoren, honen azpimultzo bat egingo da, galdera espazialek soilik osaturiko instantziak hartuz, ikusteko, espezifikoki arrazonamendu espazialean nolako eragina duen.

Gainera, beste bi transformer multimodalekin konparatuko da VisualBERT, LxMERT eta ViLTekin, hauek integratzen dituztelako *spatial embeddingak* hasieratik, eta beraz, ondorioak ateratzen lagundu dezakeelako konparaketa honek.

Esperimentazio eta konparaketaren ondoren, hainbat ondorio aterako dira: lehenik, ikusiko da, *spatial embeddingek* ez dutela diferentziarik suposatzen VQA atazan VisualBERT erabiltzerakoan. Gainera, honen arrazoia, ziurrenik, sareen aurre-entrenamendua dela argudiatuko da, informazio espaziala erabiltzen ikasteko *fine-tuning* fasea nahikoa ez dela ondorioztatuz. Horiek horrela, etorkizunerako hainbat ikerketa proposamen egingo dira, *spatial embeddingak* hobeto erabiltzen ikasteko helburuarekin.

Gaien aurkibidea

Irudien aurkibidea

Taulen aurkibidea

KAPITULUA 1

Sarrera

Adimen artifizialaren arloan, multimodalitate deitzen zaio mota desberdinetako datuak aldi berean prozesatu eta erabiltzeari. Multimodalitate honen barruan, oso ohikoa da testua eta irudia aldi berean prozesatuta ebatzi daitezkeen problema edo atazak aurkitzea. Adibidez, *visual question answering* (VQA) bezalako atazak izango genituzke, zeinetan irudi baten gaineko galdera bati erantzutea den helburua; edo *image captioning*, irudi bat hartuta oinarritzat, irudi honi testuzko deskribapen egoki bat sortzea izanik helburua. Beste adibide konplexuago bat izan daiteke *visual commonsense reasoning* (VCR), zeinetan irudi baten inguruko galdera bat erantzuteaz gain, erantzun hori aukeratzearen arrazoia eman behar den. Ataza multimodalak oso garrantzitsuak dira adimen artifizialean, nahiz eta gizakiok oso modu errazean egiten ditugun, ordenagailu bitartez zailagoak direlako ebazteko.

Horiek horrela, ataza multimodalak ebazteko, azkenaldiko joera transformerrak erabiltzea izan da, hauek baitira emaitza onenak lortzen dituztenak mota honetako atazetan. 2017. urtean, "Attention is all you need" [5] artikuluan aurkeztu zen lehen aldiz transformer arkitektura. Arkitektura honen inguruan ikertuz eta hau garatuz, hainbat motatako sarrerak aldi berean prozesatu ditzakeen transformer multimodala garatu zen beranduago, ataza multimodaletan erabiltzeko. Transformer multimodal hauen artean aurki daitezke, esaterako, LxMERT [6], ViLT [8], VisualBERT [7] edo ViLBERT [10].

Transformer multimodaletan eredu berriak sortuz zuzenean lan egiteaz gain, hauen analisiak ere egin dira. Esaterako, [11] artikuluan transformer multimodal hauen inguruko hainbat aspektu aztertzen dituzte: ea sarrera iturri bikoitzak edo banakakoak hobe funtzionatzen duen, aurre-entrenamenduko atazen eta entrenamenduko datu multzoen eragina, hiperparametroen eta hasieratze pisuen eragina, *fine-tuning* fasearen eragina eta *embedding* geruzen garrantzia.

Hala ere, orain arteko transformer multimodalen analisian ez da arreta berezirik jarri posizio *embeddingetan*. Alegia, orain arte, gehienbat, irudiak kodetzeko objektu detektatzaileak erabili dira, irudi bakoitzean objektuak eta beraien ezaugarriak detektatu eta kodetzeko. Kodetze prozesu honetan, detektatutako irudiko objektuen posizioa kodetzearen garrantzia eta forma desberdinen eragina ez da aztertu. Horregatik, lan honen helburua objektuen posizioaren inguruko informazioaren eragina aztertzea izango da, informazio hori kodetzeko modu desberdinak konparatuz, hauen eragina neurtzeko.

Horretarako, VisualBERT transformer multimodala hartu da oinarritzat, zeinak ez duen objektuen posiziorik kodetzen, nahiz eta objektu detektatzaile bat erabili. Horiek horrela, posizioa kodetzeko modu ezberdin batzuk inplementatu dira, eta elkarren artean konparatu eta aztertu. Hau egiteko, probak VQA atazan egin dira, posizio *embedding* desberdinen eragina aztertuz bertan.

VQA ataza aukeratu da, batez ere, gaur egun ataza honetan erabiltzen den datu multzoa (VQA v2.0 [4]) handia delako, eta atazaren ebaluazioa ondo definitua dagoelako metrika espezifikoekin, konparaketa erraztuz. Gainera, ataza honetan, irudien gaineko galderak egiterakoan, posizio *embeddingek* garrantzia izan dezakete, galdera batzuetan zuzenean objektuen posizioei erreferentzia egiten zaielako, esaterako, 'Zer dago ezkerreko katiluan?' edo 'Zertan ari da eskuineko pertsona?' bezalako galderetan. Horrela, bereziki arrazonamendu espaziala [1] aztertzea interesatzen denez, ebaluatzeko datu multzotik azpimultzo bat ere egin da, azpimultzo espaziala deitu zaiona, eta bereziki posizioarekin erlazioa duten galderak soilik biltzen dituena, bertan konparaketa gehiago egiteko.

Azterketaren ostean lortu diren emaitzekin hainbat ondorio atera izan ahal dira. Lehenik, ikusi da posizioa kodetzeko modu ezberdinen artean ez dagoela inolako alderik VQA atazan, VisualBERT transformerra erabiltzerakoan. Posizioaren gaineko informaziorik gabeko sistemak eta posiziodun kodeketa desberdinen artean ez da desberdintasunik lortu, ez VQA datu multzo originalean, ez eta arrazonamendu espaziala bereziki aztertzeko erabili den azpimultzo espazialean ere.

Argudiatu da, emaitza hauen arrazoia transformerrek duten aurre-entrenamendua izan daitekeela. Beste transformer batzuek aurre-entrenamendutik posizioaren informazioa erabiltzen dute, eta baliteke aurre-entrenamendu honetan informazio erabiltzen ikasteko gai izatea transformerrak. Horregatik, aurre-entrenamenduetarako erabiltzen diren atazetan, posizioaren informazioa erabiltzen ikasteko, atazak aukeratu edo diseinatu daitezke, transformerrek arrazonamendu espazialean emaitza sendoagoak lortzea nahi bada.

[1] Arrazonamendu espazial deitu zaio transformerrek irudiko objektuen posizioa ulertu eta erabiltzeko duten gaitasunari, alegia, informazio espaziala erabiltzeko gaitasunari.

KAPITULUA 2

Aurrekariak

2.1 Sarrera

Adimen artifizialean eta ikaskuntza sakonean (ingelesez *deep learning*) oso ohikoak izan dira datu mota konkretu bat sarrera gisan hartzen duten teknikak. Esaterako, irudiak, testua edota bideoa sarrera bezala erabiltzen dituzten sareak eta aplikazioak hainbat lekutan aurki daitezke. Hala ere, azken urteetan multimodalitatea indar handia hartzen aritu da. Alegia, datu mota desberdinak konbinatuta prozesatzen dituzten teknikak gero eta gehiago erabiltzen dira, errealitateko problema batzuei aurre egiteko.

Horretarako arkitektura desberdinak erabili badaitezke ere, transformerrek gero eta garrantzi handiagoa hartu dute alderdi honetan ere, eta multimodalitatearekin konbinatuz sortu dira transformer multimodalak. Horien bitartez, hainbat ataza desberdinentzat soluzioak proposatu dira, hala nola, *Visual Question Answering* (VQA) atazarako. Horietako bat izan da master amaierako lan honen aztergaia, VisualBERT transformer multimodala. Jarraian, aipatutako alderdietan gehiago sakonduko da, azalpen zehatzagoak emanez, eta baita gaur egungo artearen egoera aztertuz ere.

2.2 Multimodalitatea

Aurrekariak aztertzeko, multimodalitatea abiapuntu egokia izan daiteke. Esan bezala, multimodalitatea deitzen zaio datu mota bat baino gehiago aldi berean aztertzen dituzten teknikak erabiltzeari. Honek, abantailak izan ditzake, alde batetik, datu mota bat baino gehiago dugulako, eta, beraz, informazio gehiago eta desberdina aztertu dezakegulako; baina, horrez gain, baita datu mota ezberdinen arteko erlazioak ikasi daitezkeelako ere. Horrela, datu mota bakarrarekin baino errepresentazio sendoagoak lortu daitezke, datu mota bateko errepresentazioak beste datu motako informazioak indartu ditzakeelako.

Adibidez, hizketaren errekonozimendua egiteko, aldi berean bideoa eta audioa prozesatzen dituzten teknikak proposatu dira [12]. Beste ohiko aplikazio multimodal bat izaten da testua eta irudia bateratzen dituena. Aldi berean testua eta irudia prozesatuz bien arteko paralelismoak ikasi ditzakete sareek.

Esaterako, hemen aurki ditzakegu LxMERT [6] edo ViLBERT [10] transformer multimodalak, aldi berean testua eta irudiak prozesatzen dituztenak. LxMERTek testua eta irudia bakoitza bere aldetik prozesatzen ditu, *self-attention* geruzak erabiliz, baina, horrez gain, bi sarreren erlazioak aztertzen ditu, *cross-attention* geruzak erabiliz. ViLBERTek ezaguna den BERT arkitektura aplikatzen du ataza multimodaletan, testuaz gain irudia eta honekiko erlazioak ikasteko.

Prozesamendu multimodalaren zailtasunen artean, "Multimodal Machine Learning: A Survey and Taxonomy"[13] artikuluaren arabera, honakoak aurki genitzake:

- Errepresentazioa: Informazio multimodala ahalik eta modu egokienean adierazi eta laburtu behar da, datu iturriak osagarriak izan daitezen, eta erredundantzia gutxitu dadin.
- Translazioa: Modalitate ezberdinetako datuak ongi itzuli behar dira modu batetik bestera. Adibidez, irudi bakarra testuz modu askotara adierazi daiteke, eta testu guztiak irudi berdinarekin lotu beharko lirateke, eta baita alderantziz ere, testu bat irudi askorekin lotu daiteke, eta lotura horiek guztiak baliozkotzat hartu.
- Lerrokatzea: Datu mota ezberdinetan ordena alda daitekeenez, bi errepresentazioetako loturak nola egin arazo bat izan daiteke. Horregatik, testuinguruan jartzeak eta urrutiko dependentziak mantentzeak garrantzi handia izan ohi du.
- Fusioa: Mota ezberdinetako datuek iragarpen ahalmen desberdina izan dezakete, eta era desberdinean aprobetxa daitezke. Garrantzitsua da ahalik eta ondoen elkartzea informazio iturri desberdin hauek.
- Elkar-ikastea (ingelesez *Co-learning*): Datu iturrien artean informazioa partekatzea ere garrantzitsua da, batak besteari lagundu ahal izateko eta datuen iragarpen ahalmena areagotzeko. Hemen sar daitezke, esaterako, *cross-attention* geruzak.

Beraz, multimodalitatea, izatez, ez da arkitektura zehatz bati lotua doan zerbait, baizik eta tipologia desberdinetako datuak aldi berean erabiltzea, elkarren arteko erlazioak aztertuz. Multimodalitatearen barruan, hainbat problema desberdin planteatu dira, ataza multimodal deritzenak. Jarraian ataza multimodal hauetako batzuk aurkeztu eta azalduko dira, testuinguru aberatsagoa emateko.

2.3 Ataza multimodalak

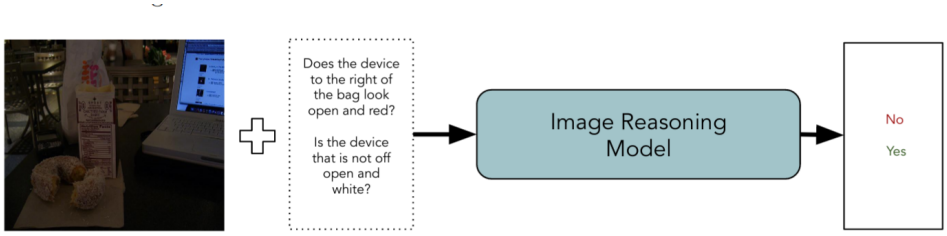
Ataza multimodalak, beraz, problema sorta bat dira, hainbat motatako sarrerak nahasten dituztenak (testua, irudiak, soinua, bideoa...) eta beraz, teknika multimodalekin tratatzea oso egokia izan daiteke. Zati honetan hainbat ataza multimodal ikusiko dira, eta laburki definitu. Ataza multimodal desberdin asko daudenez, testua eta irudiak konbinatzen dituzten ataza multimodaletan zentratuko da lana, aztertuko den VisualBERT transformerra ataza multimodal mota horretarako erabiltzen baita.

Are gehiago, batez ere, *visual question answering* atazan sakonduko da, izan ere, VisualBERTen gainean egindako azterketa, nahiz eta ataza gehiagotarako balio duen, VQA atazaren ingurukoa izan da. Hauek horrela, jarraian ikus daitezke hainbat ataza multimodal, eta haien laburpenak.

2.3.1 Natural language for visual reasoning (NLVR)

Ataza honetan, irudi bat eta honen gaineko esaldi bat dira sarrerak, eta esaldia zuzena edo okerra den zehaztu behar da. Ataza honetarako oso ohikoak dira NLVR [14] eta NLVR2 [15] datu multzoak. Lehenengoak sintetikoki eratutako 92.244 irudi eta esaldi bikote ditu, eta bigarrenak 107.292 instantzia ditu, baina benetako argazkiz osatuak. [1]

Argiago ulertzeko, 2.1 diagraman ikus daiteke NLVR atazaren prozesuaren azalpen grafiko bat.

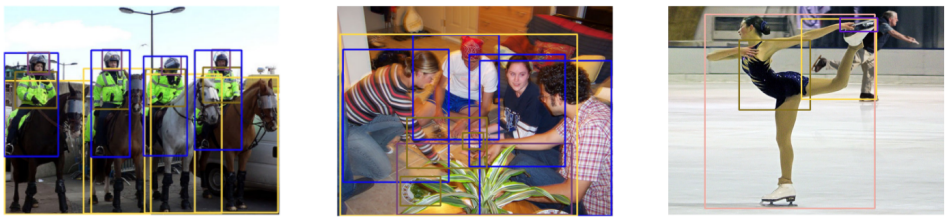


2.1 Irudia: Natural language for visual reasoning atazaren azalpen diagrama. Irudi bat eta honen gaineko esaldi bat hartuta, esaldia zuzena edo okerra den zehaztu behar da. Iturria: [1]

2.3.2 Region-to-phrase grounding

Sarrera bezala irudi bat eta honen azpititulua emanda, azpitituluan eta irudian agertzen diren entitateak elkarrekin lotzea da helburua. Alegia, deskribapenak "pertsona bat zaldi gainean" badio, zaldia eta zaldi horren gainean dauden pertsona identifikatu eta 1 eta 3 hitzekin lotu beharko lirateke. Ohiko datu multzoen artean daude Flickr30k [16] eta Visual Genome [17]. [2] [3]

2.2 diagraman ikus daitezke ataza honen hainbat adibide. Kolore bidez adierazita dago nola elkartzen diren azpitituluko hitzak irudiko objektuen *bounding boxekin*.



2.2 Irudia: Region-to-phrase grounding atazaren azalpen diagrama. Irudi bat eta honen azpititulua hartuta, testuko eta irudiko entitateak elkartu behar dira. Iturria: [2]

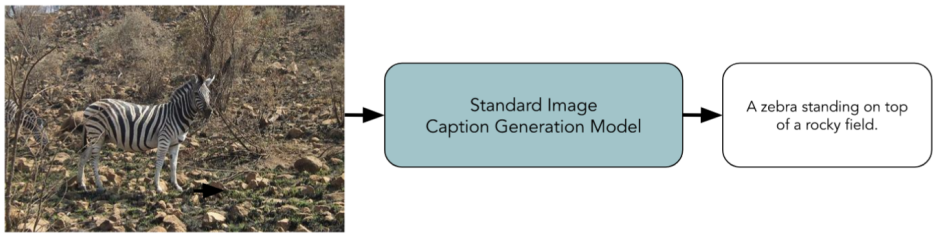
[1] NLVR eta NLVR2 datu multzoen gaineko informazioa eta gaur egungo sailkapena hemen aurki daiteke: https://lil.nlp.cornell.edu/nlvr/

[2] Flickr30k datu multzoaren inguruko informazio gehigarria hemen aurki daiteke: https://bryanplummer.com/Flickr30kEntities/, eta baita GitHubeko orrian ere: https://github.com/BryanPlummer/flickr30k_entities

[3] Visual Genome datu multzoaren inguruko informazio gehigarria hemen aurki daiteke: https://visualgenome.org/

2.3.3 Image captioning

Ataza honetan, irudi bat emanda, irudi horren deskribapena sortzea da helburua. Ohiko datu multzoen artean daude Microsoft COCO captions [18] edo Conceptual Captions [4] [19] datu multzoak. Jarraian, 2.3 diagramak azaltzen du image captioning ataza.



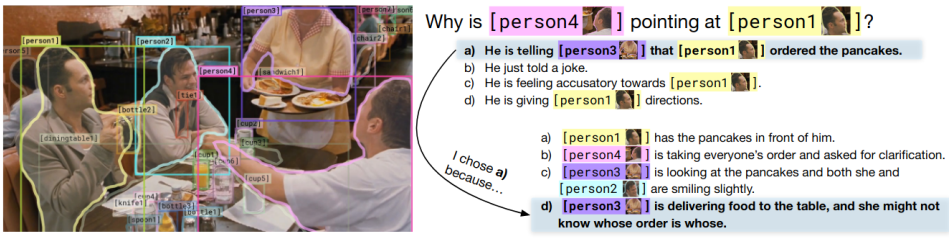
2.3 Irudia: Image captioning atazaren azalpen diagrama. Irudi bat sarreratzat hartuta, honen deskribapen bat sortu behar da. Iturria: [1]

2.3.4 Visual commonsense reasoning (VCR)

Visual commonsense reasoning ataza multimodal konplexu bat da, irudi bat eta honi buruzko galdera bat emanda, galderari ondo erantzutea da helburua. Baina, galdera hau konplexua izan ohi da, irudiaren gaineko jakinduriaz gain, munduari buruzko jakinduria orokorra behar baita galderei erantzuteko, irudian gertatzen ari dena ulertzeko, adibidez. Gainera, erantzuna emateaz gain, atazak bigarren zati bat du, eta bertan emandako erantzunaren arrazoia aukeratu behar da.

Ataza honetarako datu sorta garrantzitsuena da VCR [5] [3], eta artikuluan aipatzen den bezala, gizakientzat bereziki zaila ez den (%90 inguruko accuracya lortu ohi da) baina konputagailuentzat (%45 inguruko accuracya) oso zaila den ataza bat da, jakinduria orokorra eta arrazionalki argudiatzeko gaitasuna behar duen ataza delako.

2.4 diagramak azaltzen du VCR atazan egin beharrekoa.



2.4 Irudia: VCR atazaren azalpen diagrama. Irudi baten gaineko galdera bat erantzun behar da lehenengo eta erantzun hori arrazonatu jarraian. Iturria: [3]

2.3.5 Visual Question Answering (VQA)

Visual question answering atazaren helburua, irudi bat eta honi buruzko galdera bat sarrera bezala hartuta, galdera horri erantzutea da. Normalean, sailkapen ataza bezala planteatu

[4] Conceptual Captions datu multzoaren inguruko informazio gehiago hemen aurki daiteke: https://ai.google.com/research/ConceptualCaptions/

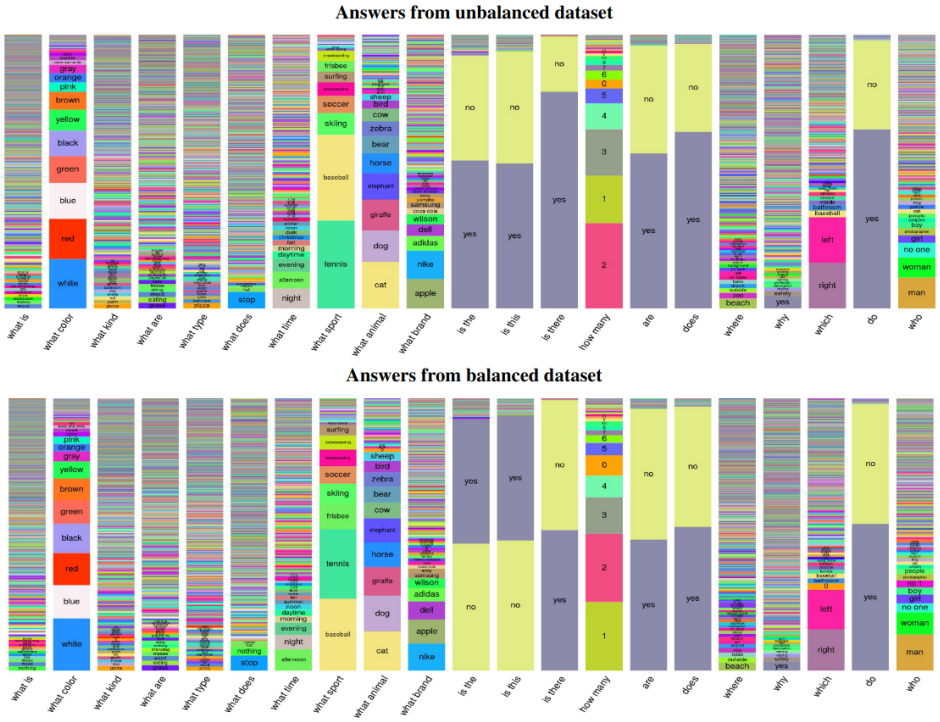
[5] VCR datu multzoaren inguruko informazio gehigarria hemen aurki daiteke: https://visualcommonsense.com/

ohi da, alegia, erantzunez osatutako hiztegi edo zerrenda bat egon ohi da, eta erantzun sorta horretatik erantzun egokiena aukeratzen saiatu behar da. Hiztegi hau osatzeko, datu multzo espezifiko bateko entrenamendu multzoan pertsonek ematen dituzten erantzunak hartzen dira, eta gehien errepikatzen diren erantzunak hiztegian sartzen dira.

Ataza honetan erabili ohi den datu multzo ohikoenetakoak VQA v1.0 [20] eta VQA v2.0 [4] dira[6]. VQA v2.0 da bertsio berri eta osatuena, eta COCO datu multzoko argazkiekin eta eszena abstraktuen argazkiekin osatua dago. 256.016 irudi biltzen ditu eta bakoitzak bere gaineko 3 galdera ditu, gutxienez (bataz beste 5,4 galdera ditu irudi bakoitzak). Gainera, galdera bakoitzarentzat 10 erantzuneko bektore bat dago, hau erabiliz, atazarako sortutako asmatze-tasa propio bat kalkulatzeko.

Datu multzo honen sorkuntza prozesuan, egileen arabera, VQA v1.0 datu multzoko hainbat errore konpontzea izan zen helburua, datu multzo orekatuago bat sortzeko. Momentuan zeuden hainbat eredu, VQA atazari erantzuteko garaian, gehiegi oinarritzen ziren galderaren zatian, erantzuna asmatzeko gai zirelarik ia irudia erabili gabe. Horregatik, datu multzo orekatuago eta handiago bat sortu zuten, benetan VQA atazan zati bisualari garrantzia emateko eta etekina ateratzeko.

Oreka hori garbi ikusten da 2.5 irudian. Lehen grafikoan VQA v1.0 datu multzoko erantzunak ikus daitezke, eta bigarrenean VQA v2.0koak. Bigarren datu multzoan erantzunen frekuentziak hobeto orekatuak daude, batez ere bai/ez motako galderetan ikusi daiteke hori.



2.5 Irudia: VQA v1.0 eta v2.0 datu multzoetako erantzunen frekuentziak. Datu multzo berriagoan erantzunak orekatuago daudela ikusi daiteke, batez ere galdera dikotomikoetan. Iturria: [4]

Oreka hau lortzeko, (irudi, galdera, erantzun) hirukote bakoitzerako (I, G, E), lehenen-

[6]VQA v1.0 eta v2.0 datu multzoen gaineko informazioa hemen aurki daiteke: https://visualqa.org/

go I' irudi bat bilatzen dute egileek, G galdera berdinari erantzuteko, baina erantzuna E' delarik, alegia, erantzun zaharra desberdina izan behar da, oreka lortzeko.

Horretarako, beraz, lehenik, irudi eta galdera bikote bakoitzarentzat beste irudi bat aurkitzen da. Antzeko 24 irudi erakusten dira, eta eskuz bat aukeratzen da, baina galdera berdinarentzat E' erantzun desberdin bat izango lukeena. Adibidez, irudi baten gaineko galdera "Zein kirol da hau?" balitz, eta erantzuna "tenisa", aukeratzen den irudi berrian beste kirol bat ikusiko litzateke, galderak zentzua izateko, baina kirol hori ezingo litzateke tenisa izan.

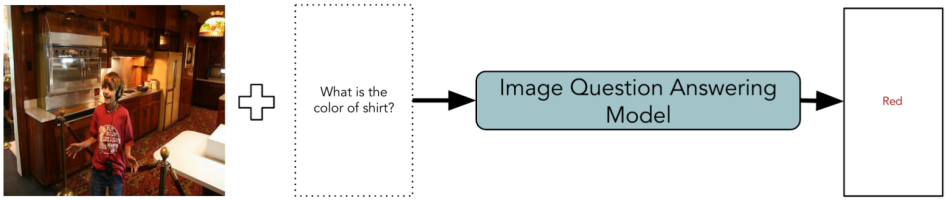
Egia da kasu batzuetan ezin dela irudi osagarririk aukeratu, galderak zentzurik ez duelako (adibidez, "Zertan ari da emakumea?" bada galdera eta irudietan emakumerik azaltzen ez bada) edo nahiz eta galdera aplikagarria izan, erantzuna berdina bada.

Irudiak lortu ondoren, beste 10 etiketatzaileri ematen zaie irudi eta galdera bikotea, eta horrela 10 erantzuneko bektore bat lortzen da instantzia bakoitzarentzat.

2.6 irudian ikusi daitezke datu multzo honetako irudien hainbat adibide, haien galdera eta erantzunekin. Ikusi daitekeen bezala, erantzunak mota askotakoak izan daitezke, objektuak, zenbakiak edo bai/ez motatakoak, esaterako. Gainera, ikusi daiteke galdera berdina duten irudi bikoteak direla, baina erantzun desberdinekoak.

2.6 Irudia: VQA v2.0 datu multzoko irudi, galdera eta erantzunen adibideak. Instantzia bakoitzak irudi bat eta honi buruzko galdera bat biltzen ditu, ondoren galdera horren erantzuna asmatzeko. Iturria: [4]

Azkenik, 2.7 diagramak erakusten du VQA ataza, nahiz eta datu multzoko adibideetan argi geratu den.



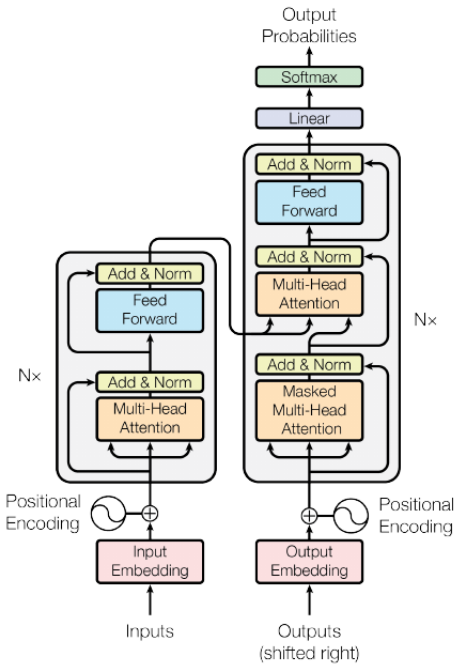
2.7 Irudia: VQA atazaren azalpen diagrama. Irudi bat eta galdera oinarritzat hartuta, erantzun egokia ematea da helburua. Iturria: [1]

2.4 Transformer multimodalak

Bestalde, multimodalitatea alde batera utziz, aipatu beharrekoa da azken urteetan transformer arkitekturak adimen artifizialean hartu duen garrantzia. "Attention is all you need"[5] artikuluan aurkeztu zen lehen aldiz arkitektura hau 2017an. Orokorrean, transformer arkitektura kodetzaile eta deskodetzaile blokez osatua dago.

Horrela, kodetzaileak sarrera bezala sinbolo sekuentzia bat jasotzen du $(x_0, ..., x_n)$, eta errepresentazio jarraitu bat sortzen du $z = (z_0, ..., z_n)$. Ondoren, deskodetzaileak irteera sekuentzia bat $(y_0, ..., y_m)$ sortzen du tarteko errepresentazio honetatik abiatuta. Pausu bakoitzean, eredua auto erregresiboa denez, sortzen duen sekuentziako elementu bakoitza erabiltzen du hurrengoa sortzeko, tarteko z errepresentazioa erabiltzeaz gain.

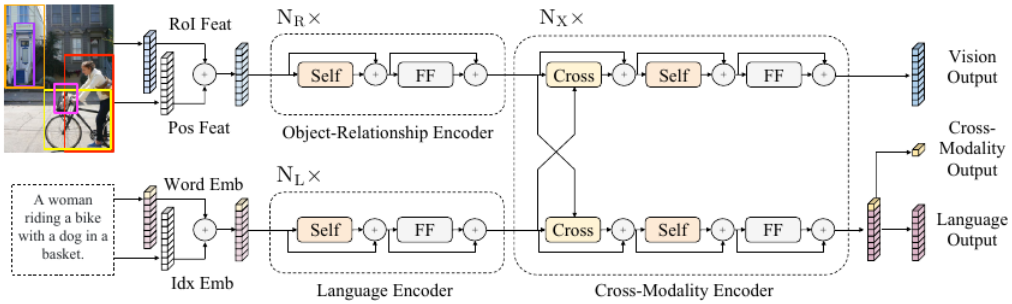
2.8 irudian ikus daiteke transformer baten arkitektura orokorra, kodetzaile eta deskodetzaile blokeekin. Ikus daitekeen bezala, bloke bakoitzak beste azpi-bloke gehiago ditu.



2.8 Irudia: Transformerraren arkitektura. Kodetzaile-deskodetzaile formako arkitektura bat da, zeinetan lehenik sarrera kodetzen den ondoren deskodetu eta irteera erabiltzeko. Iturria: [5]

Behin transformerraren arkitektura gainetik ikusi ondoren, berriro gogoratu behar da multimodalitatearen ideia, eta bi kontzeptuak batu, transformer multimodala lortzeko. Transformer multimodaletan, ideia orokorra da, nahiz eta gero eredu bakoitzak bere arkitektura propioa izan, input mota bakoitzarentzat transformer arkitektura bat izatea, eta ondoren, bi modalitateak elkartzen dituen zati bat edukitzea.

Adibidez, lehen aipaturiko LxMERT [6] transformer multimodalaren arkitektura erabili daiteke kontzeptu hau ilustratzeko. 2.9 irudian LxMERT arkitekturaren bloke kodetzailea ikus daiteke, sarrera bezala irudi bat eta esaldi bat hartzen ditu transformer honek. Sarearen funtzionamenduan sartu gabe, garbi ikus daiteke 3 bloke kodetzaile dituela sareak: horietako bi datu mota bakarrean espezializatuak dira (*object-relationship encoder* irudientzat eta *language encoder* testuentzat), eta hirugarren bat bi modalitateak konbinatzeko erabiltzen du (*cross-modality encoder*).



2.9 Irudia: LxMERTen bloke kodetzailearen arkitektura. Hiru azpi-bloketan banatuta, irudia eta testua bi bloke desberdinetan kodetzen da, eta ondoren bi modalitateak nahasten dituen hirugarren bloke batetik igarotzen da sarrera. Iturria: [6]

Horrela, kodetzaile honek sarrerak eraldatu eta 3 irteera sortzen ditu: *vision output*, *language output* eta *cross-modality output*. Jarraian, lan honen aztergaia izan den VisualBERT [7] sakonago aztertuko da, eta baita lanean zehar konparaketak egiteko erabili diren beste bi transformer ere, ViLT [8] eta LxMERT [6].

2.4.1 VisualBERT

Multimodalitatea eta transformerrak ulertu ondoren, eta ondorioz, transformer multimodalak aztertu ondoren, lan honen aztergaia izan den transformer multimodala azalduko da: VisualBERT.

VisualBERT transformer multimodal bat da, aldi berean irudia eta testua prozesatzeko prestatua dagoena. Hainbat ataza multimodaletarako prestatua dago, alegia aplikazio orokorreko teknika bat da, nahiz eta, lan honetan, batez ere VQA atazan aztertuko den. Hala ere, artikulu originalean, VQAz gain, VCR (*visual commonsense reasoning*), NLVR (*natural language for visual reasoning*) eta Flickr30k (*region to phrase grounding*) datu multzoetan neurtu zen, momentuko eredu onenak gaindituz.

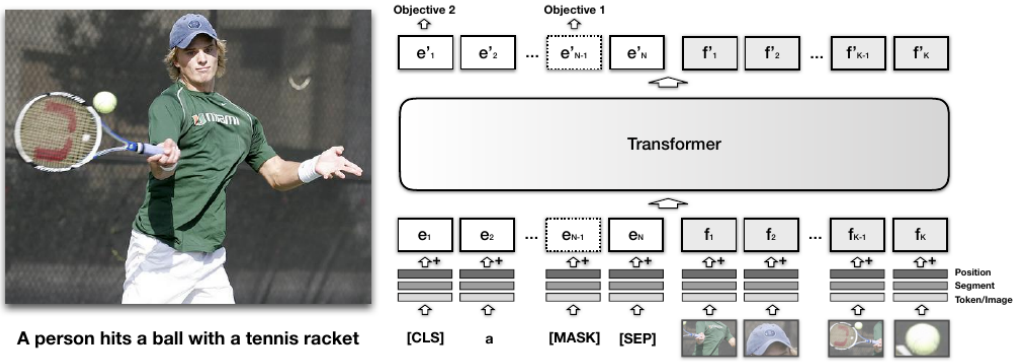
VisualBERTen ideia orokorra, transformerraren atentzio mekanismoa erabiltzea da, testua eta irudiko zatiak lerrokatzeko. BERT [21] transformerraren elementuak erabiltzeaz gain, ezaugarri bisualak erabiltzen dituzte irudiak adierazteko. Ezaugarri bisual hauek objektu detektatzaile baten bitartez lortzen dira; zehazki, Faster R-CNN [22] objektu detektatzailea erabiltzen du VisualBERTek.

Horiek horrela, irudi bateko objektu bakoitza *embedding* gisan adierazten da. *Embedding* hauetako bakoitzak hiru osagai ditu: objektuaren ezaugarriak kodetuta biltzen dituen 2048 dimentsioko bektore bat, f_o (objektua zer den eta nolakoa den kodetzen du), objektu detektatzaileak sortua; segmentu *embedding* bat, f_s, irudi bat dela adierazteko, eta ez testu bat; eta posizio *embedding* bat, f_p, irudiko eta testuko zatiak bat datozenean adierazi ahal izateko. Posizio *embedding* hau VCR atazan soilik erabiltzen da, eta ez da nahastu behar gero aipatuko diren posizioaren kodeketekin. Geroago ikusiko da, esperimentazioan irudien informazioa kodetzeko posizioaren kodeketa desberdinak erabili direla lanean zehar. Horiek ez dira f_p *embedding* hauek.

Bestalde, testuari dagokionez, esan bezala, BERT transformerraren egitura erabiltzen du, eta azken horrek hartzen duen sarrera berdina hartzen du; testu tokenizatu bat. Alegia, testua tokenizatzaile batetik pasatzen da, eta zenbaki bidezko errepresentazio bihurtzen da, hiztegi baten arabera. VisualBERTen funtzionamendu orokorra ikusi ondoren, transformer honen arkitektura ikusiko da hurrenik.

2.4.1.1 Arkitektura

VisualBERTen arkitektura 2.10 irudian ikusi daiteke. Ezkerreko aldean sarrera ikusi daiteke, irudi batek eta bere deskribapenak osatua. Sarrera hauek guztiak transformer bakar batera sartzen dira, eta irteera bat sortzen dute. Esan beharra dago arkitektura hau oso orokorra dela, izan ere, erabiltzen den ataza bakoitzerako aldaketak egiten baitizkiote. Hala ere, VisualBERTen transformerrak 12 geruza, 768ko tamaina ezkutatua (ingelesez *hidden-size*) eta 12 autoatentzio buru (ingelesez *self-attention head*) ditu. Aipagarria da, azkenik, atazaren arabera, VisualBERTen azken blokea aldatu egiten dela, bete beharreko atazara moldatzeko.



2.10 Irudia: VisualBERTen arkitektura. VisualBERTek sarrera bezala testu bat eta irudi bat hartzen ditu, eta biak kodetzaile batean sartzen ditu [SEP] token batek banaturik. Iturria: [7]

Adibidez, VQA atazarako, transformerraren azken blokea bloke sailkatzaile bat izaten da. Bloke honek [CLS] tokena hartzen du sarrera gisan, eta bi geruza linealen ostean, 3129 dimentsioko probabilitate bektore bat sortzen du, probabilitate altueneko erantzuna aukeratzeko hiztegi batetik, eta hori izaten da sarrerako galderarentzat hartzen den erantzuna.

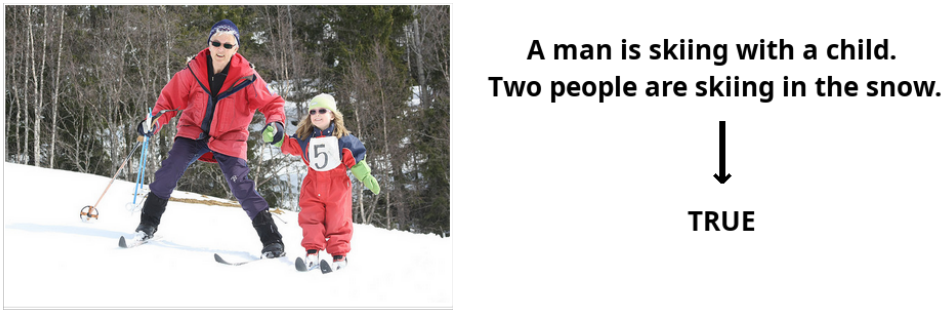
2.4.1.2 Entrenamendua

VisualBERT entrenatzerako garaian, artikuluaren egileek hiru faseko entrenamendua deskribatzen dute: *task-agnostic pretraining*, *task-specific pretraining* eta *fine-tuning*:

- *Task-agnostic pretraining*: Entrenamenduaren lehen fasean, irudia eta testua batera-tuta, bi aurre-entrenamendu egiten dira:
  - *Masked language modeling* irudiarekin: Testuko hitz batzuk estaltzen dira, irudian zati horiek estali gabe, eta helburua hitz horiek asmatzea da, irudiaren laguntzarekin. Hurrengo 2.11 irudian ikusi daiteke aurre-entrenamendu ataza honen adibide argigarriago bat.

2.11 Irudia: Masked language modeling atazaren adibidea. Irudi bat eta bere deskribapena bat hartzen dira sarreratzat, eta deskribapenean estalita dauden hitzak asmatu behar dira irudiaren laguntzaz.

  - *Sentence-image prediction*: Irudi bat eta honen bi deskribapen hartzen dira oinarritzat. Bi deskribapenetako bat irudiaren deskribapen egokia izaten da, eta besteak %50eko probabilitatea du deskribapen egoki bat izateko eta %50ekoa ez joateko irudiarekin bat. Ataza honen helburua da bigarren esaldi hau egokia den edo ez den desberdintzea. 2.12 irudian ikusi daiteke ataza honen adibide bat.



2.12 Irudia: Sentence-image prediction atazaren adibidea. Irudi bat eta bi deskribapen hartuta, bi deskribapenak irudiarekin bat datozen edo ez asmatzea da helburua.

- *Task-specific pretraining*: Ataza espezifiko baterako entrenamendua egin aurretik, beste aurre-entrenamendu fase bat egiten da, ataza espezifikoaren datu sorta erabiliz berriro *masked language modeling* irudiarekin aurre-entrenamendua errepikatzen da.

- *Fine-tuning*: Ataza espezifiko bakoitzerako azken entrenamendua *fine-tuning* fasea izaten da, ataza espezifiko horretan emaitzak maximizatzeko. Izan ere, kontuan izan behar da, artikulu originalean VisualBERT sarea 4 ataza desberdinetan probatzen dela.

2.4.1.3 Atazak

Esan bezala, VisualBERT irudia eta testua konbinatzen dituzten ataza desberdinetan erabili daiteke; zehazki, 4 atazatan egin dituzte esperimentuak egileek: VQA, VCR, NLVR eta *region-to-phrase grounding*.

- *Visual question answering*: Ataza honetarako sailkapen problema gisan planteatzen da ataza, nahiz eta teorian erantzunek irekiak izan behar luketen. Orduan erantzun multzo batetik erantzun probableena aukeratzea da egiten dena.

- *Visual commonsense reasoning*: Bi azpi atazatan banatzen dute ataza hau. Lehenengo, galdera eta irudia erabiliz erantzuna lortzen du ereduak, 4 erantzun posibleko multzo batetik egokia aukeratuz. Ondoren, emandako erantzuna arrazonatu behar da, eta horretarako, galdera eta aurreko pausuko erantzuna sarrera bezala hartuta, 4 arrazonamendu posibleetako bat aukeratu behar du ereduak.

- *Natural language for visual reasoning*: Ataza honetan egin beharrekoa irudi baten gaineko esaldi bat zuzena edo okerra den desberdintzea da, eta beraz, sailkapen ataza bezala planteatzen da berriro. Kasu honetan, ordea, VQAn ez bezala, erantzuna ez da hiztegi zabal batetik hartzen, bi emaitza posible soilik daudelako: bai/ez.

- *Region-to-phrase grounding*: Azken ataza honetan, lehen esan bezala irudiko objektuen *bounding boxak* eta testuko hitzak elkartu behar dira. Horretarako, *self-attention* bloke gehigarri bat ipintzen dute egileek. Bloke barruko *attention head* bakoitzaren batez besteko pisuak neurtzen dituzte, eta *bounding box* bakoitzarentzat, atentzio handiena lortzen duen azpi-hitza hartzen dute iragarpen bezala.

Azken finean, atal honekin adierazi nahi dena da VisualBERT ataza desberdin askotan erabili daitekeela eta momentuan zegoen artearen egoera hobetu edo berdintzen zuten emaitzak lor zitzakeela. Hala ere, kontuan izan behar da lan honetan VisualBERTen gainean egingo den analisia VQA atazaren gainean soilik izango dela.

2.4.2 LxMERT eta ViLT

LxMERT [6] eta ViLT [8] beste bi transformer multimodal dira, irudi eta testuarekin lan egin dezaketenak, eta VQA atazari aurre egiteko gai direnak. Geroago, esperimentuen 4 atalean, bi transformer horiekin konparatuko da VisualBERT. Horregatik, aurrekarien atal honetan labur deskribatuko dira bi transformer berri hauek.

VisualBERTen antzekoak dira bi transformerrak, baina oinarrizko diferentzi garrantzitsu bat dute: aurre-entrenamendutik hasita, posizioaren informazioa gehitzen dute, bakoitzak bere erara.

2.4.2.1 LxMERT

LxMERT [6] transformer multimodal bat da, VisualBERTen antzera, irudi bat eta galdera bat sarreratzat hartzen dituena. Alabaina, VisualBERTekin funtsezko diferentzia batzuk ditu, bai transformerraren egituran eta baita sarreren *embeddinga* egiterako garaian ere, tartean posizioaren informazioaren tratamenduan.

Testuaren *embeddinga* egiteko, hitz mailako kodeketa egiten du sareak, $(w_1, ..., w_n)$, *WordPiece* tokenizatzailea erabiliz. Ondoren, hitzak esaldiaren barruan duen posizioarekin batera, bi bektoreen proiekzioa egiten da, ondoren, batuta, posizioaren informazioa barne duen *embedding* osatuago bat lortuz. Aipatzekoa da, baita ere, normalizazio geruza batetik pasatzen dela azken *embeddinga*.

Irudiari dagokionez, konboluzio-sare bat erabili beharrean, objektu detektatzaile bat erabiltzen du, VisualBERTek bezala. Horrela m objektu detektatzen dira $(o_1, ..., o_m)$, bakoitzarentzat 2048 dimentsioko ezaugarri bektore bat eta objektuaren posizioa erabiliz. Kasu honetan, posizioa kodetzeko, objektuaren posizioa definitzen duten 4 zenbaki erabiltzen dira, objektuaren goi-ezkerreko puntua (x_0, y_0) eta behe-eskuineko puntua (x_1, y_1) erabiliz. Ezaugarri bektorea eta posizio bektorea proiektatu eta batu egiten dira, azkenik normalizazio geruza batetik igaroz.

Ikerlarien hitzetan, posizioaren informazioa oso garrantzitsua da objektuen kodeketan, batez ere, *masked object prediction* aurre-entrenamendu atazan, baina baita ere, efektiboa dela neurtu dutelako arrazonamendu espazialerako. Ondoren, esperimentuen 4 atalean aurkeztuko diren emaitzetan ikusiko da baietz, badirudiela posizioaren gaineko arrazonamendua ikasteko gai dela sarea.

Embeddingaz gain, sarearen kodeketa blokeak ere oso interesgarriak dira: atal honetan lehenago aipatu den bezala, eta 2.9 irudian ikusi denez, 3 bloke kodetzaile ditu sareak. Horietako bik (*Object-relationship encoder* eta *Language encoder*) modalitate indibidualak kodetzen dituzte, alegia, irudia eta testua, bakoitza bere aldetik, eta hirugarrenak, bi modalitateen arteko erlazioak kodetzen ditu (*Cross-modality encoder*).

LxMERTen outputari dagokionez, honek hiru irteera sortzen ditu: *language*,*vision* eta *cross-modality* outputak. Transformer bat izanik, kontuan izan behar da sarearen irteera sarrera eraldatua izango dela, eta, beraz, testu bidezko sarrera *language output* bihurtuko da eta irudien sarrera *vision output*. *Cross-modality outputa* lortzeko, bestalde, testu sarrerari [CLS] token berezi bat eransten zaio, eta hau eraldatuta, lortuko da *cross-modality outputa*.

Entrenamenduari dagokionez, laburki, egileen helburua sareak bi modalitateen arteko erlazioak ikastea denez, hizkuntzaren modalitateko, irudien modalitateko eta modalitate konbinatuko aurre-entrenamenduak egiten dituzte. Esaterako, *masked language modeling* irudiarekin, VisualBERTen aurre-entrenamenduan ere badagoena, edo *masked object prediction*, irudiaren zatiak asmatzean datzana, testuaren laguntzaz.

2.4.2.2 ViLT

ViLT [8] beste transformer multimodal bat da, VisualBERTen eta LxMERTen antzekoa baita ere, baina irudiak prozesatzerako garaian beste oinarrizko desberdintasun handi bat daukana.

Egileen artikuluan aipatzen da transformer multimodalek irudia prozesatzerako garaian erabiltzen dituzten teknikengatik (objektu detektatzaileak edo neurona-sare konboluziona-

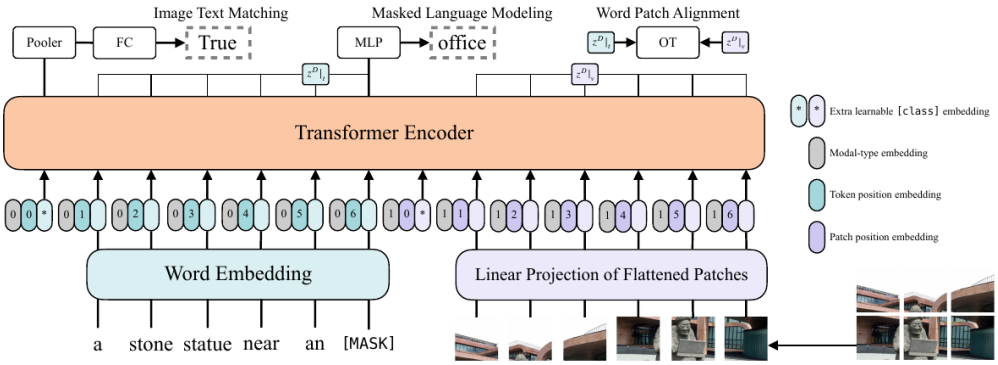
lak) arazoak izan ditzaketela. Alde batetik, konputazio aldetik pisutsuak dira, objektuen ezaugarriak kalkulatzea detektatzaileak erabiliz motela delako interakzio multimodalekin alderatuz; eta bestalde, sarearen espresio maila mugatu egiten da, objektu detektatzailearen mugen barnean ibili behar delako nahitaez, azken horren hiztegitik ezin baita atera.

Horregatik, egileek ViLT proposatzen dutenean, *embedding* bisuala egiteko forma guztiz aldatzen dute. Irudia 32×32 zati berdinetan egiten dute, eta zati bakoitza proiekzio linealeko geruza batetik pasatzen da. Horrez gain, zatiaren posizioaren inguruko informazioa ere batzen zaio irudiaren *embeddingari*.

Matematikoki adieraziz, irudia ($I \in \mathbb{R}^{C \times H \times W}$) zatitan egiten da, eta zati bakoitza bektore bihurtzen da $v \in \mathbb{R}^{N \times (P^2 \cdot C)}$ dimentsionalitatearekin. Ondoren, proiekzio lineala egin eta posizioaren proiekzioa gehitu ondoren, $\overline{v} \in \mathbb{R}^{N \times H}$ lortuko litzateke. [7]

Embedding hau [23] artikuluan aurkeztu zen lehen aldiz, eta izugarri txikitzen du irudia konputatzearen denbora, proiekzio linealak oso eraginkorrak direlako konputazionalki. Azken finean, testuaren *embeddinga* egiten denean ere, proiekzio lineal bat egin ohi da, eta beraz, bi sarreren *embeddingen* konputazio kostua oso antzekoa da. Aipatzekoa da, gainera, 2.4 milioi parametro behar direla proiekzio hauek egiteko.

Horiek horrela, 2.13 irudian ikusi daiteke transformerraren arkitektura. Esan bezala, eskuinean ikus daitekeen irudiaren prozesamendua zuzenean irudi zatien proiekzio lineala da, eta testuaren kasuan, VisualBERT eta LxMERTek bezala, BERTen tokenizatzailea erabiltzen dute. Esaldi tokenizatuko hitz bakoitzari hitzak esaldi barruan duen posizioa gehitzen zaio, proiektatu ondoren batzeko.



2.13 Irudia: ViLT transformerraren arkitektura. Irudia eta testua batera hartzen ditu sarreratzat ViLTek. Irudia prozesatzeko, zatika banatu eta zatien proiekzio linealak egiten ditu. Iturria: [8]

Testuaren ($t \in \mathbb{R}^{L \times |V|}$) *embeddinga* egiterakoan, proiekzioa egin ondoren, eta posizioaren proiekzioa batu ondoren, $\overline{t} \in \mathbb{R}^{L \times H}$ lortuko litzateke[8].

Sarearen outputa ere ikus daiteke irudian. LxMERTen kasuan irteerak sinplifikatuago azaltzen baziren ere, oso antzekoak dira ViLTen irteerak ere, azken finean, output bisuala, testu outputa eta [CLS] tokenak baitira sarearen irteerak, hau ere transformer bat izaki.

Entrenamenduari dagokionez, ViLT ere hainbat atazatan aurre-entrenatu da. Zehazki,

[7] C: irudiaren kanalak, H: altuera, W: zabalera, P: (P, P) zati bakoitzaren erresoluzioa, N: $N = HW/P^2$

[8] L: sekuentziaren luzera, V: tokenizatzailearen hiztegiaren tamaina, H: transformerraren tamaina ezkutua, kasu honetan 768

bi ataza aukeratu dira horretarako: *image text matching* eta *masked language modeling*. Ataza hauetan zehar, hitz osoak ezkutatzen dira (eta ez hitz zatiak, nahiz eta tokenizatzaileak hala funtzionatzen duen). Bestalde, fine-tuning fasean zehar *image-augmentation* teknikak erabiltzen dira, RandAugment [24], zehazki. [9]

[9] RandAugment tekniketako bi politika ez dira aplikatzen: kolore alderantzizkatzea (kolorearen inguruko galderak egon daitezkeelako) eta ebaketa (irudiaren zati garrantzitsuak desagerrarazi ditzakeelako, hala nola objektu konkretuak).

KAPITULUA 3

Metodologia

3.1 Sarrera

Zati honetan lanaren muina aztertuko da. Lan honen helburua, transformer multimodalek arrazonamendu espaziala egiterakoan, objektuen posizioen kodeketa desberdinen eragina ikustea da. Horretarako, oinarritzat VisualBERT transformerra hartu da, VQA atazarekin eta VQA v2.0 datu multzoarekin batera. Alegia, objektuen posizioaren informazioa kodetzeko erak eragiten al du ataza honetan lortzen dituen emaitzetan? Eta hala bada, zein kodeketa izango litzateke erabilgarriena ataza ahalik eta ondoen egiteko?

Horiek horrela, atal honetan esperimentazioan erabiliko diren hainbat tresna eta zehaztasun azalduko dira. Lehenik, erabiliko diren posizio kodeketa ezberdinak azalduko dira, nola funtzionatzen duten azalduz, eta ondoren transformerraren sarrera bezala nola erabiliko diren esplikatuz.

Horrez gain, esperimentazioan hainbat proba egingo direnez, proba horietan erabiliko diren datu multzoak deskribatuko dira. Jarraian, erabiliko diren ebaluazio metrikak azalduko dira, VQA atazan erabiliko diren asmatze-tasa eta galera funtzioa (ingelesez *loss function*) aurkeztuz, eta nola kalkulatzen eta erabiltzen diren azalduz. Azkenik, azalpen teorikoen ostean, inplementaziorako erabili diren erreminta eta hardwarearen deskribapen labur bat egingo da.

3.2 VisualBERTen funtzionamendua

Orduan, lehenik eta behin, VisualBERTek funtzionatzeko behar dituen sarrerak, eta ondoren sortuko dituen irteerak nolakoak diren azaldu behar da, tartean duen prozesua ulertzeko. Horretarako, datu sortako instantzia bakoitza definitzea komeni da, lehenik, ikusi ahal izateko nola eraldatzen duen instantzia hau transformerrak. Instantzia bakoitzak irudi bat eta galdera bat ditu, sarrera bezala erabiltzeko, eta hamar erantzuneko bektore bat emaitza bezala.

Lehen aipatu bezala, ordea, irudia ez da zuzenean irudi bezala kodetzen, alegia ez dira pixelen balioak erabiltzen. Horren ordez, objektu detektatzaile bat erabiltzen da (Faster R-CNN) eta irudian dauden objektuak detektatu eta hauek errepresentatzeko ezaugarri-

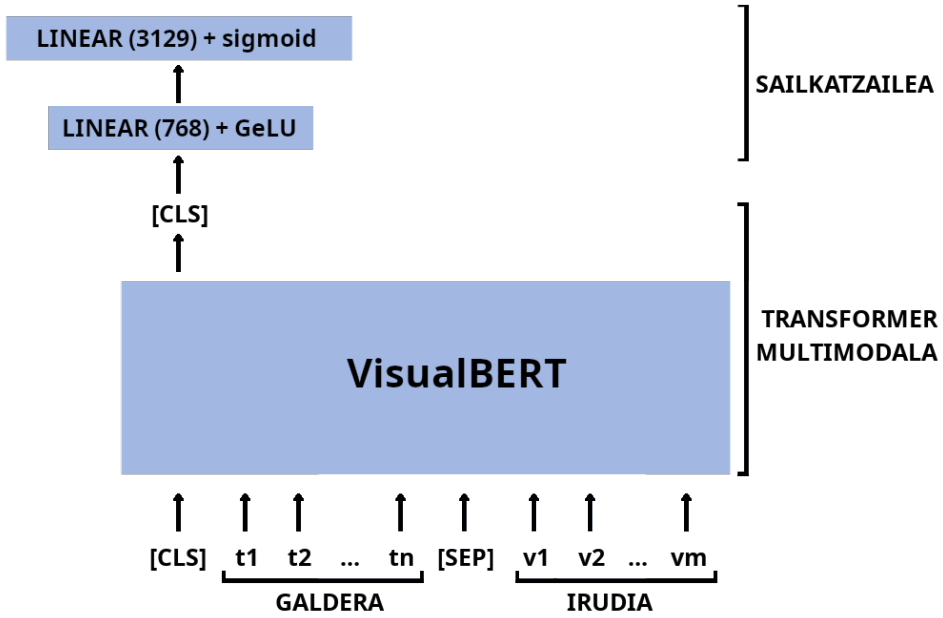
bektoreak sortzen dira. Irudiko objektu bakoitza 2048 zenbakiko bektore batekin errepresentatzen da, eta irudiko, gehienez, 36 objektu biltzen dira.

Badira irudia prozesatzeko objektu detektatzaileak erabiltzen dituzten transformer gehiago ere. Hauetako batzuek, detektatzen dituzten objektuak errepresentatzeko ezaugarri-bektoreak sortzeaz gain, objektu hauek irudiaren barruan duten posizioa ere kontuan hartzen dute, eta transformerrari sarrera bezala ematen diote. VisualBERTek, ordea, ez du hau egiten, objektu detektatzaileak sortutako ezaugarri-bektorea soilik erabiltzen du.

Alegia, adibidez, irudian txakur beltz bat agertzen bada, objektu detektatzaileak detektatzen badu ere, txakur hau irudiaren ezkerrean egon edo eskuinean, errepresentazioa berdina izango litzateke. Horregatik, esan bezala, lan honetan egingo dena izango da, VisualBERT oinarritzat hartuz, objektuen posizioa ere sarreratzat erabiltzea, alegia, transformerrari aldaketa batzuk egitea, ikusteko ea posizioaren informazioa emanda, nola aldatzen diren emaitzak. Horiek horrela, objektuak irudian duen posizioa eta tamaina kodetzeko, jarraian azalduko den bezala, 2 kodeketa desberdin erabili dira kasu honetan.

Sarrerako galderari dagokionez ere, testu bezala kodetu beharrean, tokenizatzaile bat erabiltzen da, oso ohikoa den bezala. VisualBERTen arkitektura propioagatik, BERT transformerraren tokenizatzailea erabiltzen da. Tokenizatzaile honek testua token-bektore bihurtzen du, hiztegi baten arabera. Alegia, hitz bakoitza zenbaki bihurtzen du, hiztegian duen posizioaren arabera. Ondoren, beste xehetasun gehiago ere badaude, adibidez, esaldien hasieran [CLS] tokena jartzen da, eta amaieran [SEP] tokena, besteak beste.

Emandako sarrera hau eraldatuko du VisualBERTek. 3.1 diagraman ikusi daiteke nola eraldatzen den sarrera VQA atazan, amaieran sartu den galderaren erantzuna lortzeko.



3.1 Irudia: VisualBERTen prozesuaren diagrama, VQA atazan. Sarreratzat galdera eta irudia erabiliz, [CLS] tokena erabiltzen da, sailkatzaile batean sartu eta erantzun bat lortzeko. Horretarako, sailkatzaileak bi geruza lineal erabiltzen ditu: lehengoak 768 tamaina ezkutua eta GeLU aktibazio funtzioa du, eta bigarrenak 3129ko tamaina eta sigmoide aktibazio funtzioa.

Behetik hasita, galdera hartzen du lehenik eta behin VisualBERTek, [CLS] token bat erantsita. Galdera hau tokenizatzailetik pasatzen da eta proiekzio lineal baten ondoren transformerrean sartzen da. Bestalde, irudia objektu detektatzailetik pasatzen da, eta proiekzio lineal baten ondoren beste sarrera bezala erabiltzen da.

Irteeran, [CLS] tokena erabiltzen da erantzuna lortzeko. [CLS] token honen gainean geruza-anitzeko pertzeptroi bat jartzen da, diagraman sailkatzailea deitu zaiona. Pertzeptroi hau bloke lineal bat da, 768 tamainako geruza ezkutua, GeLU aktibazio funtzioa eta 3129 tamainako irteera geruza bat dituena. Geruza honen ostean, beraz, 3129 tamainako probabilitate bektore bat lortuko da. Bektore honi sigmoide funtzioa aplikatuko zaio, probabilitate altueneko aukera hartzeko. Aukera honek hiztegi batetik erantzuna hartuko du, eta hau izango da VQA atazan behar den emaitza.

Hiztegi hau datu sorta bakoitzari lotua izaten da, alegia, datu sorta bakoitzak bere erantzunen hiztegia du; eta VQA v2.0ren kasuan hiztegi honen tamaina 3129 erantzunekoa da. Hiztegi hau osatzeko, datu multzoko entrenamenduko partizioan gizakien erantzunetatik 10 aldiz edo gehiago errepikatzen diren erantzunak biltzen dira.

Jarraian, esperimentuetan erabili diren posizio kodeketak azalduko dira. Alegia, objektu detektatzaileak identifikatzen duen objektu bakoitzari posizioaren inguruko informazioa eransteko erabili diren bi metodoak: *rectangle encoding* eta *grid encoding*.

Bi kodeketa hauez gain, esperimentuak egiteko oinarrizko VisualBERT eredu bat ere erabili da, posizioaren inguruko inolako informaziorik erabiltzen ez duen oinarri bat. Horregatik, irudiaren errepresentazioa egiteko objektu detektatzaileak lortutako ezaugarri bektorea soilik hartzen da, eta posizioaren inguruko informazioa alde batera uzten da. Oinarrizko eredu hau, beraz, sinpleena da, eta, batez ere, posizioak ezertarako balio ez duela ondorioztatzen bada, ondorio hori indartzeko erabiliko da esperimentuetan.

3.2.1 Rectangle encoding

Irudien errepresentaziorako, VisualBERT originalean, lehen esan bezala, objektu detektatzaile batek sortutako ezaugarri bektoreak erabiltzen dira, baina ez zaio posizioaren informaziorik eransten, nahiz eta objektu detektatzaileak informazio hau ere ematen duen, objektu bakoitzari *bounding box* bat esleitzen baitio.

Horregatik, bektore hauetako bakoitzari posizioaren informazioa gehitu ahal zaio, eta horrela, irudiaren barruan objektuen posizioa jakin daiteke. Rectangle encoding kodeketan, objektu bakoitzaren posizioa sei zenbaki erabiliz osatzen da: x_0, x_1, y_0, y_1, w, h. Zenbaki hauekin, objektuaren goi ezkerreko izkina (x_0, x_1), behe eskuineko izkina (y_0, y_1), zabalera (w) eta altuera (h) kodetzen dira. VisualBERTek erabiltzen duen Faster R-CNN objektu detektatzaileak informazio guzti hau ematen du, beraz, oso erraza da inplementatzeko eta eransteko.

3.2 irudian ikusi daiteke COCO datu sortatik ateratako irudi baten adibide bat eta bere kodeketa. Kasu hipotetiko honetan, irudian su hidrante gorri bat ikus daiteke, eta hau bilduz, bere tamaina eta posizioa dituen lauki berde bat marraztu da. Karratu honen (*bounding box*) tamaina eta posizioa dira rectangle encoding kodeketan ur hidrantearen ezaugarri-bektoreari erantsiko zaizkionak. Laukiak honako balioak izango lituzke beraz: $x_0 = 397, y_0 = 182, x_1 = 441, y_1 = 267, w = 44, h = 85$. [1]

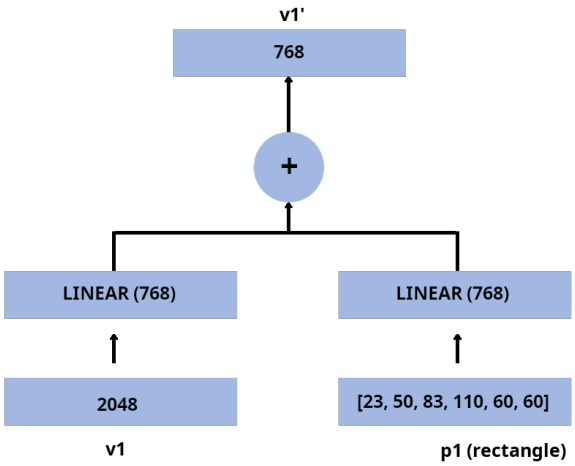
[1] Kontuan izan behar da, nahiz eta pixel balio absolutuak erabili diren adibidea sinpleago azaltzeko, izatez

3.2 Irudia: *Rectangle encoding* adibidea. Irudiko objektu bat kokatzeko bere *bounding boxaren* koordenatuak erabiltzen dira, goi-ezkerreko puntua, behe-eskuinekoa, altuera eta zabalera.

Sei zenbaki hauek, normalizatu ondoren, 768 dimentsioko bektore batera proiektatzen dira; ondoren, ezaugarri bektorea ere 768 dimentsioko bektore batera proiektatzen da, eta bi bektoreak batzen dira. Horiek horrela, irudi bakoitzeko objektu bakoitzaren kodeketa $f_o \in \mathbb{R}^{768}$ honela adierazi daiteke, ezaugarri bektorea $f_v \in \mathbb{R}^{2048}$ eta posizio bektorea $f_r \in \mathbb{R}^6$, 768 dimentsiotako bektoretara proiektatu ondoren:

$$f_o = f_{vp} + f_{rp} \text{ non } f_o \in \mathbb{R}^{768}, f_{vp} \in \mathbb{R}^{768}, f_{rp} \in \mathbb{R}^{768}$$

Kasu honetan f_{vp} ezaugarri bektore proiektatua izango litzateke, eta f_{rp} posizio bektore proiektatua. 3.3 diagramak azaltzen du nola batzen zaion posizioa ezaugarri bektoreari, bi bektoreak proiektatu ondoren.



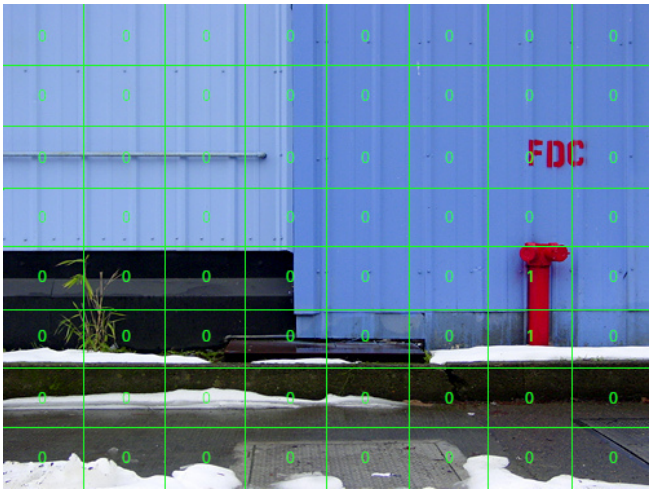
3.3 Irudia: *Rectangle encoding* kodifikazioan, lehenik posizio bektorea eta ezaugarri bektorea proiektatzen dira, eta ondoren biak batuta lortzen da azken errepresentazioa.

balio erlatiboak edo normalizatuak erabiltzen direla, irudiaren tamainara doituz, balio guztiak 0 eta 1 tartera normalizatuz.

3.2.2 Grid encoding

Grid encodingean posizioa kodetzeko, lehenik, irudia $n \times n$ tamainako lauki-sare batean banatzen da. Ondoren, objektuak okupatzen dituen laukietan 1 bat jarriko litzateke, eta beste lauki guztietan 0, bi dimentsiotako matrize bat lortuz. Azkenik, matrize honen ilarak bata bestearen atzetik jarriko lirateke, $1 \times n^2$ tamainako bektore bat lortzeko.

3.4 irudian ikusi daiteke 8×8 tamainako lauki-sare bat oinarritzat erabiliz nola kodetuko litzatekeen lehen erabilitako adibide berdina. Kasu honetan, su hidrantea dagoen laukietan (2 lauki) 1a jarriko litzateke, eta beste guztietan 0. Ondoren, 8×8 dimentsioko matrizea lautu eta 1×64 tamainako bektore bat lortuko litzateke, 39 eta 47 garren posizioetan 1ak izango lituzkeena: $f_g = \{0, 0, ..., 1, 0, ..., 1, 0, ..., 0\}$.



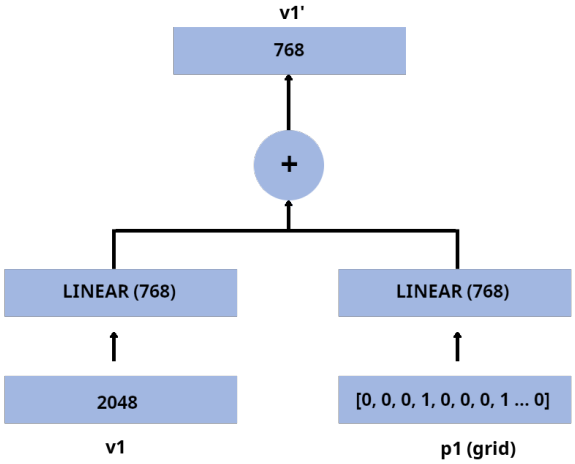
3.4 Irudia: 8×8 *Grid encoding* adibidea. Objektuen posizioa kodetzeko, irudia lauki-sare batean banatzen da, eta ondoren objektua dagoen laukietan 1 bat ipintzen da, eta beste lauki guztietan 0.

Horiek horrela, tamaina desberdinetako lauki-sareak erabili daitezke esperimentu desberdinetarako, gero eta sare handiagoa, orduan eta zehaztasun handiagoa lortuko litzateke. Azkenik, rectangle encodingaren kasuan bezala, $1 \times n^2$ tamainako bektore hau 768 floateko bektore batera proiektatuko litzateke, ezaugarrien bektore proiektatuari batzeko.

Matematikoki formalizatzeko, kasu honetan ezaugarri bektorea $f_v \in \mathbb{R}^{2048}$ proiektatu eta grid posizio bektorea $f_g \in \mathbb{R}^{n^2}$ proiektatu eta batu behar dira:

$$f_o = f_{vp} + f_{gp} \text{ non } f_o \in \mathbb{R}^{768}, f_{vp} \in \mathbb{R}^{768}, f_{gp} \in \mathbb{R}^{768}$$

Bigarren kodeketa honetan, f_{vp} ezaugarri bektore proiektatua izango litzateke berriro, eta f_{gp} posizio bektore proiektatua. 3.5 diagramak erakusten du kasu honetan, nola egiten den proiekzioa eta batura azken errepresentazioa lortzeko.



3.5 Irudia: *Grid encoding* kodifikazioan ere, lehenik posizio bektorea eta ezaugarri bektorea proiektatzen dira, eta ondoren biak batuta lortzen da azken errepresentazioa.

3.3 Datu sortak

Azkenik, erabiliko diren transformerra eta posizioaren kodeketa desberdinen gaineko xehetasunak ikusi ondoren, esperimentuetan erabiliko diren datu multzoak azaldu eta deskribatuko dira.

VQA atazan lan egingo denez, erabiliko den datu sorta nagusia VQA v2.0 izango da, artikulu originalean bezala. Ondoren, ulermen espazialaren inguruko azterketa egingo denez, VQA v2.0 datu multzotik azpimultzo bat aterako da, azpimultzo espaziala deituko zaiona. Horrez gain, esperimentuen atalean "Visual Spatial Reasoning" [9] artikuluko emaitza batzuk konparatuko direnez, bertan proposatutako VSR datu multzoaren gaineko azalpenak ere emango dira. Hala ere, VSR datu multzoa ez denez zuzenean esperimentu propioetan erabili, geroago aurkeztuko da, esperimentuen 4 atalean bertan.

3.3.1 VQA v2.0

Erabiliko den datu multzo nagusia, VQA v2.0, 2 kapituluan deskribatu bada ere, era zehatzagoan azalduko da zati honetan, esperimentuetan erabili diren partizioak esplikatuz, adibidez. Beraz, datu multzo honek 256.016 irudi biltzen ditu, horietatik 204.721 COCO datu sortakoak, eta besteak eszena abstraktuak. Horrez gain, irudi hoien gaineko 1.105.904 galdera ditu, entrenamendu/balidazio/test partizioetan banatuta, eta galdera bakoitzak 10 erantzuneko bektore bat.

Test partizioa ez denez domeinu publikokoa, entrenamendu eta balidazio partizioak erabili dira sarea entrenatu eta bitartean balioztatzeko, eta emaitzak balidazio partizioaren gainean lortu direnak izango dira, entrenamendu osoa bukatu ostean. Izatez, ebaluazio metodologia hau ez litzateke egokia izango ereduak ebaluatzeko, baina, kasu honetan VisualBERT bere buruarekin konparatuko denez, emaitzak beren artean konparagarriak izango dira.

Horiek horrela, VQA v2.0 ren kasuan, entrenamendu partizioak 443.757 galdera ditu, eta horiei lotutako irudi eta erantzunak; eta balidazio partizioak 214.354 galdera, dagozkien

irudi eta erantzunekin.

3.3.2 VQA v2.0 azpimultzo espaziala

Lanaren helburu nagusia galdera espazialetan VisualBERTen emaitzak neurtzea denez, VQA v2.0 datu multzotik azpimultzo bat egitea erabaki da, esan bezala. Azpimultzo espazial honetan, entrenamendu partizioa berdin mantendu da, sarea era berdinean entrenatu ahal izateko, baina balidazio partiziotik ez dira galdera guztiak aukeratu, espazioarekin edo posizioarekin zerikusia dutenak bakarrik. Horrela, sarearen ahalmena neurtzea pentsatu da, galdera espazialei soilik erantzuterako garaian.

Hau egiteko, balidazio partizioko galdera guztiak hartu dira, eta hitz espazialen zerrenda bat [2]. Zerrenda honek posizioaren gaineko hainbat hitz edo hitz multzo ditu, esaterako, *left of*, *above*, *facing away from*, eta baita objektuen tamainari buruzkoak ere, adibidez, *high*, *huge* edo *little*. Ondoren, balidazio partizioko galdera guztiak abiapuntutzat hartu eta filtratu egin dira: galderak hitz espazialen zerrendako elementuren bat badu barnean, hartu egin da, eta bestela baztertu.

Hitz espazialen zerrenda hau egiteko, bi iturri desberdin erabili dira. Lehenik eta behin, "Visual Spatial Reasoning" [9] artikulutik atera dira 64 hitz. Artikulu honetan, irudien analisian gaur egun erabiltzen diren teknika ezberdinak neurtzen dituzte, arrazonamendu espazialaren atazan. Horretarako, datu multzo bat proposatzen dute, arrazonamendu espazialaren atazan teknikak neurtzeko. Horrela, datu multzo honen galderetan 64 erlazio espazial proposatzen dituzte, hala nola, *behind*, *above* edo *next to*, eta bertatik hartu dira hitzak.

Horrez gain, datu azpimultzoa osatuagoa izan dadin, hitz espazialen beste zerrenda bat bilatu da, eta filtratu, tamainaren, posizioaren eta orientazioaren inguruko hitz gehiago bilatzeko[3]. Zerrenda honekin, aurreko hitzen zerrenda osatu da, azkenean balidazio partizioa filtratzeko. Horiek horrela, galderen azpimultzo espazial honekin, entrenamendu partizioa ez da aldatu, baina balidazio partizioak 89.971 galdera ditu.

3.4 Ebaluazio metrikak

Esperimentuak egiterako garaian, ereduak entrenatzeko eta emaitzak neurtzeko bi metrika erabili dira: asmatze tasa eta galera.

3.4.1 Asmatze tasa

Erabili den asmatze tasa metrika VQA atazarako proposaturiko asmatze tasa bat da. Hau kalkulatzeko, lehenik kontuan izan behar da datu sortak gizakiek emandako 10 erantzuneko bektore bat duela. Horrez gain, kontuan izan behar da baita ere, sarearen irteera erantzun bakarra dela, eta hori erabiliko dela asmatze tasa kalkulatzeko.

Asmatze tasa kalkulatzeko, beraz, honakoa egin behar da: 10 erantzuneko bektorearen posizio bakoitzeko, beste 9 erantzunak hartuko dira, zuzenak direnen kopurua kontatu, eta zati 3 egin. Ondoren, lortzen den zenbakiaren eta 1 zenbakiaren arteko minimoa aukeratuko

[2] Azpimultzoa sortzeko erabili diren hitz guztien zerrenda 5 eranskinean ikus daiteke

[3] Hitz gehigarriak lortzeko erabili den zerrenda hemen aurkitu daiteke: https://www.columbusschoolforgirls.org/uploaded/Spatial_Awareness/Spatial_Vocabulary.pdf

da. Horrela, berriro 10 zenbakiko bektore bat lortuko da, eta bektore honen bataz bestekoa izango da erabiliko den asmatze tasa metrika.

Adibidez, demagun, sareak galdera hipotetiko baterako eman duen erantzuna "Bai" dela, eta datu multzoak ematen duen erantzun bektorea honakoa dela: g = ["Bai", "Bai", "Bai", "Bai", "Bai", "Bai", "Bai", "Bai", "Ez", "Ez"]. Horiek horrela, asmatze tasa metrikaren emaitza honela kalkulatuko litzateke: lehenik zenbakien bektorea lortuko genuke, $score_{vec}$=[1, 1, 1, 1, 1, 1, 1, 1, 1, 1], eta ondoren bataz bestekoa kalkulatu, $acc = 1$.

Aldiz, sareak emandako emaitza "Ez" izan balitz, $score_{vec}$=[0.6, 0.6, 0.6, 0.6, 0.6, 0.6, 0.6, 0.6, 0.3, 0.3] izango litzateke, eta honen bataz bestekoa eginaz, $acc = 0,54$. Honela adierazi daiteke asmatze tasa formula baten bitartez, kontuan izanik n erantzun kopurua dela:

$$acc(ans) = \frac{\sum_{i=1}^{n} min(1, \sum_{j=1}^{i-1} ans == g_j + \sum_{k=i+1}^{n} ans == g_k)/3)}{n}$$

3.4.2 Galera

Galera funtzioari dagokionez, erabiliko dena entropia bitar gurutzatua logitekin (ingelesez *binary cross entropy with logits*) izango da, Pytorch liburutegian zuzenean inplementatuta dagoena, *BCEwithlogitsloss* klasean. Metrika honek entropia bitar gurutzatua (ingelesez *binary cross entropy*) eta *sigmoide* funtzioak konbinatzen ditu. Sailkapen bitarretan erabili ohi da entropia bitar gurutzatua galera funtzioa.

Pytorchen inplementazioaren dokumentazioaren arabera, honela irudika daiteke formula, N batcharen tamaina izanik:

$$\ell(x, y) = L = \{l_1, \ldots, l_N\}^\top, l_n = -w_n[y_n \cdot log\sigma(x_n) + (1 - y_n) \cdot log(1 - \sigma(x_n))]$$

Ikus daitekeen bezala, y_n helburu bektoreko n-garren elementua izango litzateke (0 eta 1 balioak har ditzake), eta x_n egin den n-garren iragarpena izango litzateke (0 eta 1 balioen artean edozein balio har dezake). Horiek horrela, iragarpen on bat izateko, $y_n = 1$ den kasuan (klase positiboa), x_n 1 zenbakitik ahalik eta hurbilen dagoen zenbaki bat izan behar da, iragarpena probabilitate indartsukoa izateko. Aldiz, $y_n = 0$ den kasuan (klase negatiboa), x_n 0 tik ahalik eta hurbilen egon behar da, negatibo indartsua izateko.

Horregatik, formulak bi zati ditu: $y_n \cdot log\sigma(x_n)$ zatiak iragarpen positiboen zatia neurtzen du, eta $(1 - y_n) \cdot log(1 - \sigma(x_n))$ zatiak negatiboena. Izan ere, helburu aldagaia 1 denean, bigarren zatia deusezten da, eta helburu aldagaia 0 denean, lehen zatia. Horrez gain, ikus daiteke bi zatietan sartuta dagoela sigmoide funtzioa eta baita eskala logaritmikoa aplikatzen dela ere.

3.5 Inplementazioa eta erabilitako tresnak

Atal honetan azaldutako xehetasun teoriko guztiak programatu eta inplementatu behar izan dira, noski, esperimentuak egin eta emaitzak aztertu ahal izateko. Horretarako, Python programazio lengoaia eta honen hainbat liburutegi erabili dira, baita konputaziorako hardware berezia ere; jarraian deskribatuko dira erabilitako erreminta garrantzitsuenak.

3.5.1 Pytorch

Pytorch ikaskuntza sakonerako erabiltzen den Python programazio lengoaiako liburutegi bat da.[4] Kode irekiko software librea da, Facebook enpresaren adimen artifizialeko laborategiak garatua.

Pytorch erabiliz, besteak beste, tentsoreen konputazio oso eraginkorra egin daiteke, txartel grafikoen konputazio ahalmena erabiltzen uzten baitu, CUDA bezalako sistemei probetxua ateraz. Horrez gain, neurona-sareak eraikitzeko ere erabili daiteke, eta baita hauek entrenatu eta probatzeko ere.

Horregatik, Pytorch oso liburutegi sendoa da adimen artifizialeko proiektuak garatzeko, eta beste hainbat liburutegiren oinarria ere bada, jarraian ikusiko den moduan. Horiek horrela, Pytorcheko hainbat metodo erabiltzeaz gain, proiektuaren pipeline orokorra liburutegi honen gainean eraiki da, erabili diren beste liburutegi batzuek ere Pytorchen erabilera egiten baitute. [5]

3.5.2 Pytorch Lightning

Pytorch Lightning kode irekiko beste liburutegi bat da, Pytorchen erabilera erraztea helburu duena. Labur esanda, Pytorchentzat maila altuko interfaze bezala funtzionatzen du, idazten den kodea irakurterrazagoa izan dadin, eta esperimentuak errazago erreproduzitu ahal izateko.

Pytorch Lightning erabiliz, erreminta arina eta eraginkorra izanik, Pytorchen funtzio konplexuagoak era errazean eta eraginkorrean erabili daitezke. Horiek horrela, lanean zehar egin diren esperimentu guztietarako oso lagungarria izan da, exekuzio desberdinak egiteko. Entrenamendu eta test prozesuak asko erraztu ditu, eta baita exekuzio desberdinen artean aldaketak egitea ere. [6]

3.5.3 Huggingface transformers

Huggingface transformers Python programazio lengoaiarako beste liburutegi bat da, transformerrekin lan egitea ahalbidetzen duena maila altuko interfaze lana eginez. Liburutegi honekin transformerrak kargatzea izugarri errazten da, ondoren erabiltzeko edo entrenatzeko.

Liburutegi honen egilea enpresa bat da, eta ereduak haren zerbitzarietatik deskargatu daitezke, aurrez entrenatuta edo entrenatu gabe, biltegi digital batetik. Transformer desberdin asko daude biltegian, eta lanean zehar aipatuko diren ereduak bertatik deskargatu dira. Gainera, eredu hauei aldaketak egitea ahalbidetzen du liburutegiak, esperimentuak diseinatu ahal izateko.

Azkenik, aipatu behar da, ereduez gain, bestelako erremintak ere badituela liburutegiak. Hala nola, transformerren sarrerak eraldatzeko tokenizatzaileak, entrenamenduetarako behar diren datu multzoak edo bestelako tresna erabilgarriak. [7]

[4] Pytorch C++ programazio lengoaiarako ere erabili daiteke, nahiz eta Pythoneko atala garatuagoa dagoen.

[5] Pytorchen inguruko informazio gehiago hemen aurkitu daiteke: https://pytorch.org/

[6] Pytorch lightningen inguruko informazio gehigarria hemen aurki daiteke: https://www.pytorchlightning.ai/

[7] Huggingface transformersen inguruko informazioa gehiago, eta baita ereduen eta tresnen gaineko informazioa hemen aurki daiteke: https://huggingface.co/

3.5.4 Tensorflow eta tensorboard

Tensorflow, Pytorchen antzera, kode irekiko liburutegi bat da, ikasketa automatikorako erabiltzen dena. Hau ere liburutegi zabala izanik, ikerketa arlo honetan behar diren hainbat erreminta biltzen ditu. Erreminta guzti horiekin, ikasketa automatikorako inguruneak sortzen laguntzen du. Horrela, eredu desberdinak eraiki, entrenatu, probatu eta ezarri daitezke.

Horiek horrela, konboluzio-sareak, transformerrak edo bestelako ereduak erabiliz sistemak eraikitzea ahalbidetzen du[8]. Eskaintzen dituen tresnetako bat, tensorboard da, lan honetan emaitzak gorde eta aztertzeko erabili dena.

Tensorboard Tensorflowen barruan integratutako bistaratze erreminta bat da. Besteak beste, ereduen entrenamenduan zehar sortzen diren metrikak eta datuak gorde eta bistaratzeko erabili daiteke. Lanean zehar, emaitzak bistaratu eta konparatzeko, hauen grafikoak egin eta entrenamendu prozesuak ondo garatu direla ziurtatzeko erabili da.[9]

3.5.5 Hardwarea

Hardwareari dagokionez, Euskal Herriko Unibertsitateko Informatika Fakultateko konputazio zerbitzariak erabili dira. Urrutiko komunikazioa erabiliz, SSH (*secure shell*) bitartez, programaturiko esperimentuak exekutatu dira zerbitzari hauetan, beraien hardwarea erabiliz. Konputaziorako erabili diren txartel grafikoak, zehazki, Nvidia Titan X eta Nvidia Titan Xp izan dira.

[8] Tensorflowen inguruko informazio gehigarria hemen aurki daiteke: https://www.tensorflow.org/

[9] Tensorboarden inguruko informazio gehiago hemen aurki daiteke: https://www.tensorflow.org/tensorboard

KAPITULUA 4

Esperimentuak eta emaitzak

4.1 Sarrera

Metodologiako 3 atalean aipatu den bezala, master amaierako lan honen helburua VisualBERTek informazio espaziala nola erabiltzen duen aztertzea da, ikusteko ea probetxurik ateratzea lortzen duen, eta hala bada, posizioaren zein kodeketa izan daitekeen eraginkorrena, *rectangle*, *grid*, edo posizioaren informaziorik ez ematea. Horretarako, VQA v2.0 datu multzoan eta honen azpimultzo espazialean egingo dira probak, datu multzo nagusian emaitzak neurtuz, eta ondoren, bereziki galdera espazialak dituen azpimultzoan.

Honez gain, beste bi transformer multimodalekin konparaketa egingo da beste artikulu batzuetako emaitzak aipatu eta konparatuz, eta hipotesi batzuk formulatuz, ikuspegi osatuago bat emateko. LxMERT [6] eta ViLT [8] transformerrak konparatuko dira VisualBERTekin, ikusteko ea informazio espaziala hobe erabiltzen duten sare hauek. Izan ere, berez, VQA atazan, VisualBERTek ez du posizioaren gaineko informaziorik erabiltzen aurreikuspenak egiteko, eta, beraz, egin daitekeen hipotesi bat da, LxMERT eta ViLTek errendimendu hobea lortu dezaketela ataza honetan.

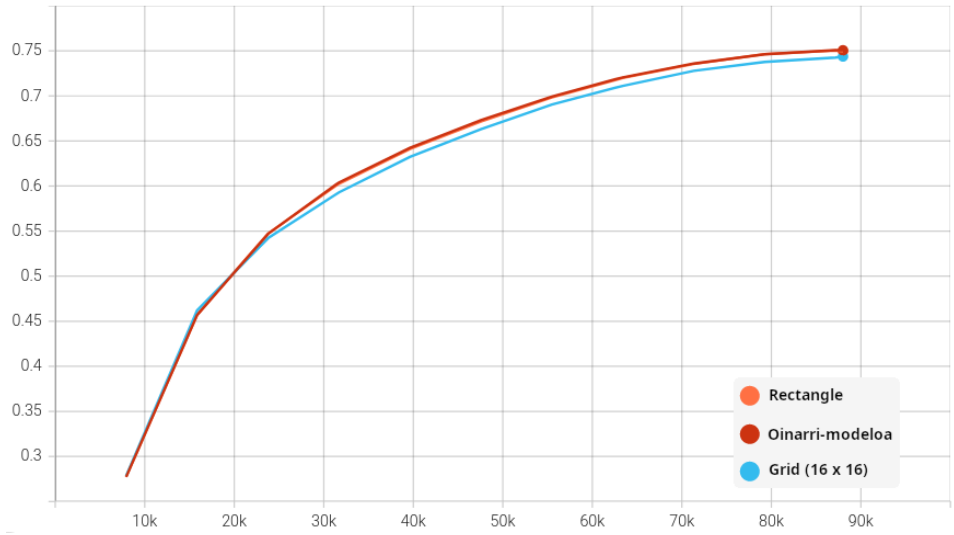
4.2 Hiperparametroen aukeraketa

Hurrengo bi esperimentuetan zehar, VisualBERTen bertsio ezberdinak elkarren artean konparatu nahi dira. Horregatik, ez da hiperparametroen doikuntzarik egin esperimentuetan, helburua ez baitzen ahalik eta emaitza onenak lortzea, baizik eta bertsio ezberdinen arteko konparaketa justua izatea, eta baldintza berdinetan egitea. Horrela, esperimentu guztietan hiperparametro berdinak erabili dira. Entrenamendurako erabili diren hiperparametroen artean, honakoak izango genituzke:

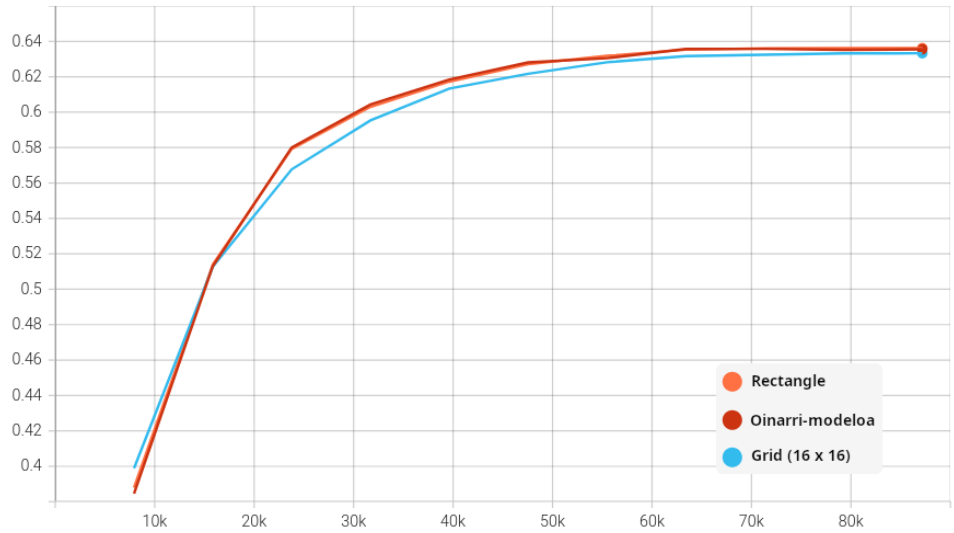
- *Batcharen* tamaina: 56
- Ikasketa-tasa (ingelesez *Learning-rate*): 5×10^{-5}
- Entrenamendu pausuak: 88.000
- *Epochak*: 11

Horrez gain, konparaketa justua egiteko, esperimentuaren exekuzio bakoitzean lortutako azken iterazioko eredua hartu da kasu guztietan. Ereduak beti azken iterazio honetan lortzen du asmatze-tasa altuena, eta beraz azken iterazio honetako ereduak konparatuko dira elkarren artean beti. 4.1 eta 4.2 grafikoek erakusten dituzte lehen esperimentuko asmatze-tasen eboluzioak, entrenamendu eta balidazio partizioetan, hurrenez hurren.

Esperimentu honetan exekuzio gehiago egin badira ere, adibidez, tamaina ezberdinetako lauki-sareak erabiliz edo pisu ezberdinak erabiliz transformerra hasieratzeko, kodeketa mota bakoitzeko exekuzio bakarra erakutsiko da, kurba guztiak oso antzekoak direlako, eta ez dutelako informazio garrantzitsurik eskaintzen.

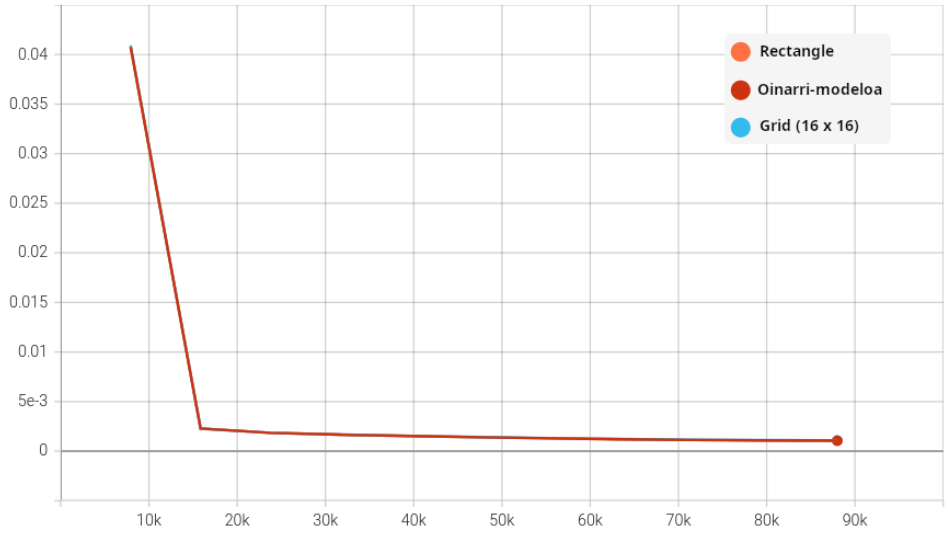


4.1 Irudia: Asmatze-tasaren eboluzioa, entrenamendu partizioan. Azken iterazioan lortzen dute eredu guztiek asmatze-tasa onena.

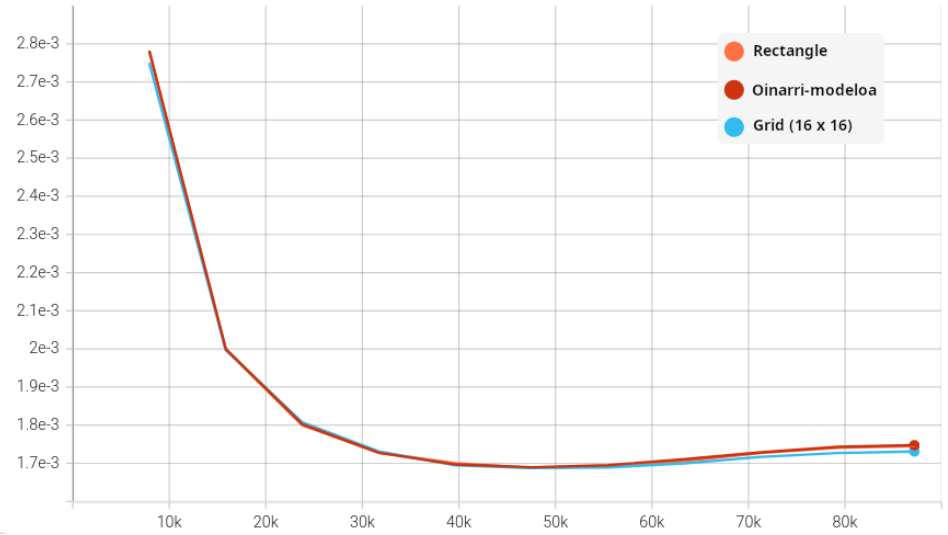


4.2 Irudia: Asmatze-tasaren eboluzioa, balidazio partizioan. Azken iterazioetan kurba zapaldu bada ere, azken iterazioan lortzen da asmatze-tasa onena.

Ikus daitekeen bezala, azken iterazioan lortzen dira asmatze tasa altuenak. Horrez gain, 4.3 eta 4.4 grafikoetan ikus daitezke galeren eboluzioak ere.



4.3 Irudia: Galeraren eboluzioa, entrenamendu partizioan. Azken iterazioan lortzen dute eredu guztiek galera baxuena.

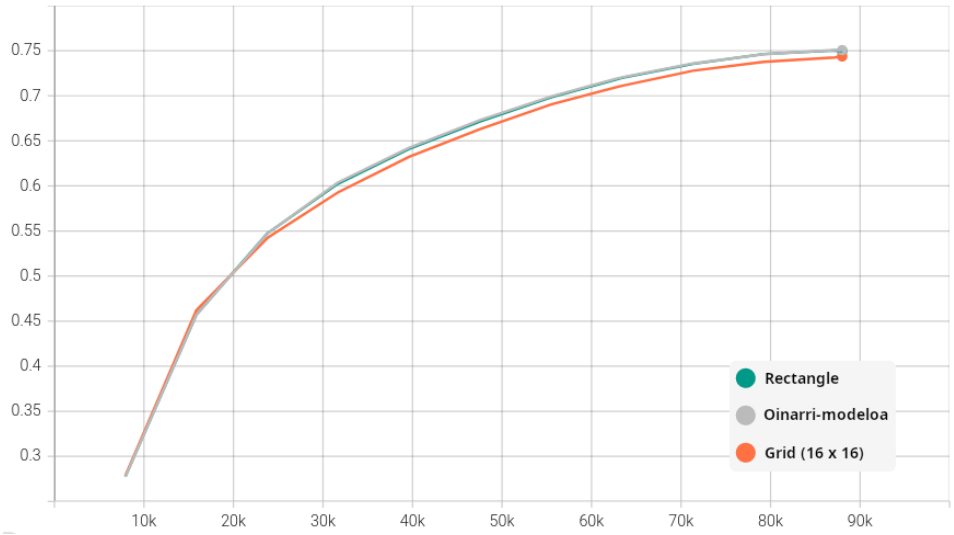


4.4 Irudia: Galeraren eboluzioa, balidazio partizioan. 50.000garren iterazio inguruan lortzen dute ereduek galera baxuena.

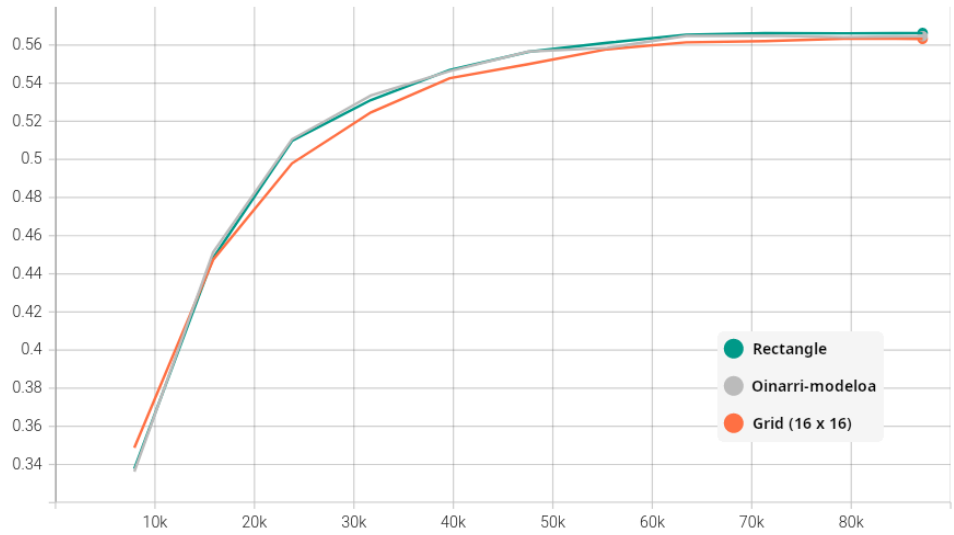
Entrenamendu kurbek, ikus daitekeen bezala, eboluzio egokiak izan dituzte: asmatze-tasa gorantz doa entrenamendu osoan zehar entrenamendu partizioan, nahiz eta balidazio partizioan 65.000garren pausuan, gutxi gorabehera, jada kurba horizontal bihurtzen den. Noski, entrenamendu partizioan asmatze-tasa altuagoak lortzen dira balidazio partizioan baino. Galeren eboluzioa ere oso antzekoa da, lehen pausuan izan ezik; bestela, 50.000garren pausuaren inguruan jotzen du behea balidazio partizioan.

Bigarren esperimentuan ere, ereduak termino berdinetan konparatu nahi izan direnez,

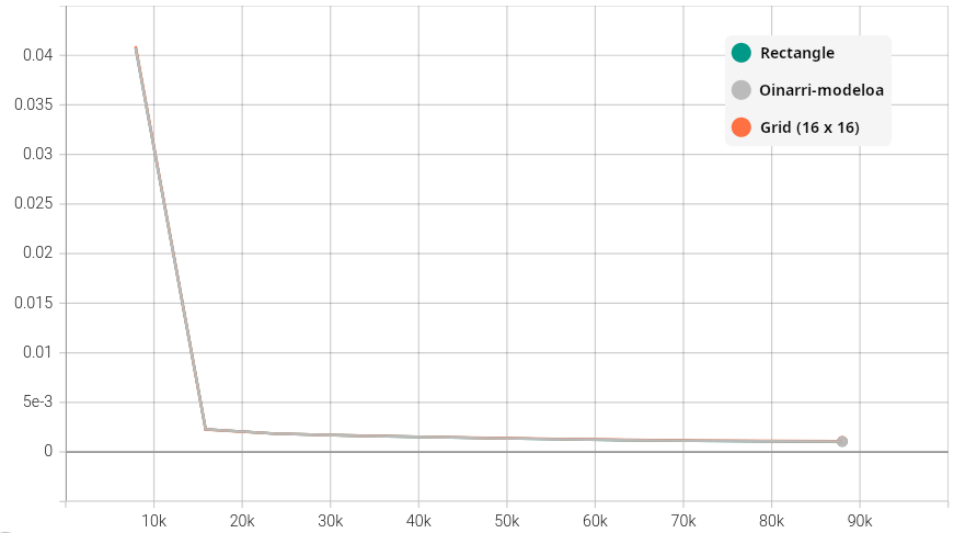
hiperparametroak ez dira aldatu, eta aukeratutako eredua beti ere entrenamenduko azken iteraziokoa izan da. Izan ere, kasu honetan ere, asmatze-tasa onenak azken iterazioan lortzen dira. 4.5 eta 4.6 grafikoek erakusten dute hori, bigarren esperimentu honetako asmatze tasen eboluzioekin, eta 4.7 eta 4.8 grafikoek galeren eboluzioak erakusten dituzte.



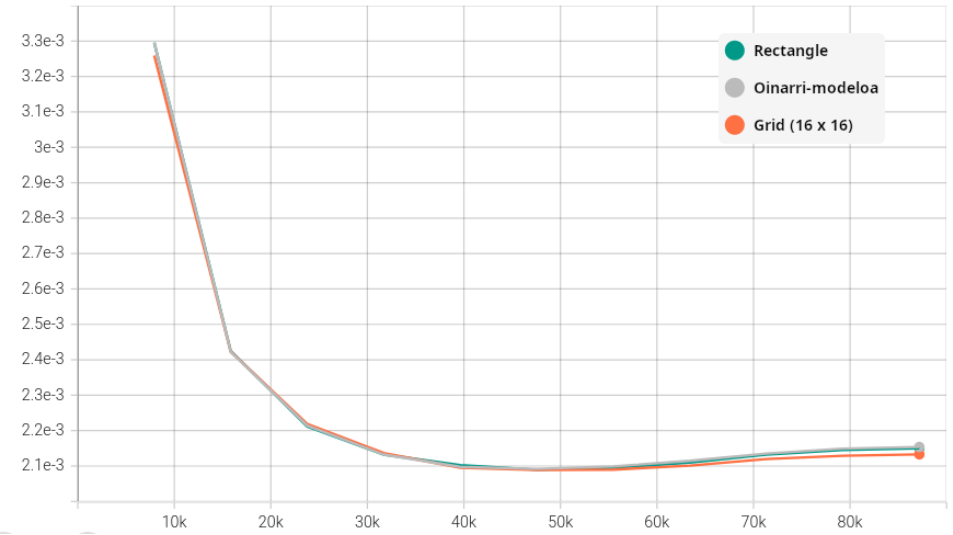
4.5 Irudia: Asmatze-tasaren eboluzioa, entrenamendu partizioan, azpimultzo espazialean. Azken iterazioan lortzen dute eredu guztiek asmatze-tasa onena.



4.6 Irudia: Asmatze-tasaren eboluzioa, balidazio partizioan, azpimultzo espazialean. Azken iterazioan lortzen dute eredu guztiek asmatze-tasa onena, nahiz eta amaieran kurba zapaldu.



4.7 Irudia: Galeraren eboluzioa, entrenamendu partizioan, azpimultzo espazialean. Azken iterazioan lortzen dute ereduek galera baxuena.



4.8 Irudia: Galeraren eboluzioa, balidazio partizioan, azpimultzo espazialean. 50.000garren iterazio inguruan lortzen dute galera baxuena ereduek.

Kasu honetan ere, asmatze-tasaren eta galeraren eboluzioak ere lehen esperimentuan bezalakoak dira. Orokorrean asmatze-tasa baxuagoak lortu dira, ikusi den moduan, baina kurbak oso antzekoak dira, bestela. Azken iterazioko ereduak konparatuko dira, beraz.

4.3 Emaitzak

4.3.1 VQA v2.0

Lehenik eta behin, VQA v2.0 datu multzoaren gainean entrenatu eta ebaluatu da VisualBERT, posizioaren kodeketa ezberdinak probatuz. Lehenago aipatu den bezala, aurkeztuko diren

emaitzak balidazio partizioan lortu diren emaitzak dira, entrenamendua amaitu ostean.

Horiek horrela, beraz, posizio kodeketa desberdinekin, 4.1 taulan ikus daitezke lortu diren emaitzak. Aipatu behar da, baita ere, entrenamendua egiten hasi aurretik, VisualBERTen pisuak hasieratzerakoan, COCO datu multzoaren gainean aurre-entrenamendua egin ondoren lortutakoak erabili direla. Aurre-entrenamendu hau 2 atalean ikusi da zehatzago, baina, labur esanda, *masked language modeling* irudiarekin eta *sentence-image prediction* atazek osatzen dute COCO datu multzoa erabiliz.

Posizioaren kodeketa	Asmatze-tasa (%)
Oinarrizko eredua	63,8
Rectangle	63,74
Grid (16×16)	63,83
Grid (28×28)	63,87
Grid (32×32)	64,04
Grid (64×64)	63,95

4.1 Taula: VQA v2.0 datu multzoaren emaitzak

Taulatik ondorioak ateratzen hasi aurretik, aipamen txiki bat egin nahi da, COCOren gaineko aurre-entrenamenduko pisuez gain, esperimentuak BERTen pisu originalekin hasieratuta ere egin direlako, eta emaitzak aurkeztu ez badira ere, esan behar da BERTen pisuen eta COCOko aurre-entrenamenduko pisuen artean ez dagoela diferentzia handirik. Beraz, badirudi COCO datu multzoaren gainean egindako aurre-entrenamenduak ez duela askorik laguntzen kasu honetan emaitzak hobetzen.

Hala ere, taulatik atera daitekeen ondorio garrantzitsuena da, posizioaren kodeketa desberdinek ez dutela inolako eraginik erakusten. Oinarrizko ereduaren, *rectangle* eta *grid* kodeketen arteko diferentziak txikiegiak dira ezer aipagarririk azpimarratzeko. Horiek horrela, kontuan hartuz, oinarrizko ereduan ez dela posizioaren inguruko inolako informaziorik ematen, esan daiteke posizioaren informazioa eman edo ez eman, berdin dela.

Alegia, VisualBERTen kasuan, VQA atazan bederen, posizioaren informazioak ez du inolako eraginik galderei forma egokian erantzuterako garaian. Posizioaren gaineko informazioa emateak ez dio laguntzen aurreikuspen hobeak egiten.

4.3.2 VQA v2.0 azpimultzo espaziala

Ikusi berri den ideia hau sakonago aztertzeko, hurrengo esperimentuan, aipatutako azpimultzo espazialean esperimentu berdinak egin dira, posizioaren kodeketa desberdinak erabiliz, berriro.

Printzipioz, badirudi, azpimultzo espazial horretan ere diferentzia handirik ez litzatekeela egon behar, datu multzo osoan ez bada diferentzia handirik egon, batez beste. Hala ere, posible da justu azpimultzo espazialean diferentzia handixeagoa izatea, beharbada, eta orduan beharbada posizioaren kodeketaren baten laguntza ikus daiteke nonbait.

Oinarrizko eredua, *rectangle* eta *grid* (16, 28, 32 eta 64 tamainakoak) kodeketak erabiliz, azpimultzo espazial honen balidazio partizioan ebaluatu dira ereduak, entrenamendu berdina egin ondoren. Alegia, entrenamendua berriro baldintza beretan egin da (datu

multzo eta hiperparametro berak), eta, beraz, eredua berdina da, baina honen emaitzak beste testuinguru batean ebaluatu dira, testuinguru espazialean gehiago sakonduz. 4.2 taulan aurkezten dira lortu diren emaitzak.

Posizioaren kodeketa	Asmatze-tasa (%)
Oinarrizko eredua	56,75
Rectangle	56,65
Grid (16×16)	56,65
Grid (28×28)	56,67
Grid (32×32)	56,94
Grid (64×64)	56,98

4.2 Taula: VQA v2.0 datu multzoaren emaitzak, azpimultzo espazialean

Azpimultzo espazialaren gaineko esperimentuetan lehen aipatutako hipotesia bete dela dirudi. Kasu honetan ere, ez da desberdintasun aipagarririk ikusten posizioaren kodeketen artean. Badirudi, berriro ere, ez dagoela diferentziarik lauki-sare tamaina desberdinen artean, ez eta *rectangle* kodeketaren eta oinarrizko ereduaren artean ere. Ondorioz, esan daiteke, azpimultzo espazialera mugatzen bagara ere, teorian posizioaren informazioak bereziki lagundu behar lukeen azpimultzoa izanda ere, ez dela hori gertatzen.

Bestalde, ikusi daiteke, baita ere, lortutako batez besteko asmatze-tasak txikiagoak direla azpimultzoan multzo osoan baino. Honen arrazoia izan daiteke azpimultzo espazialeko galderak erantzuteko zailagoak izatea, eta beraz, VisualBERT ez izatea gai erantzunak horren erraz emateko. Bestela, esan nahiko luke galdera espazialak erantzuten bereziki txarra dela VisualBERT, ez galderak zailagoak direlako, baizik eta informazio espaziala behar bezala ikasteko ez delako gai.

Bigarren hipotesi honetan esaten dena, alegia, informazio espaziala erabiltzeko zailtasunak izatea, garbi uzten du esperimentuak berak. Posizioaren kodeketa desberdinen artean ez dago diferentzia argirik, eta beraz, informazio espaziala ez da erabiltzen. Hala ere, garbi utzi behar da azpimultzo espazialeko emaitzak batez beste 7,1 puntu okerragoak izatearen arrazoia beharbada ez dela hau.

Lehen bi esperimentu hauekin, beraz, argi eta garbi gelditu da VisualBERT ez dela gai posizioaren informazioa erabiltzeko, nahiz eta azken *fine-tuning* fasean era desberdinetara eman informazio hau. Baina zergatik izan daiteke hau?

Kontuan izan behar da VisualBERTen aurre-entrenamenduan ez dela posizioaren informazioarekin lanik egiten. Alegia, esperimentu hauetan VQA atazan azken *fine-tuning* fasean soilik sartu da posizioaren informazioa, eta badirudi ez dela nahikoa posizioaren inguruko informazioa ongi erabiltzen ikasteko. Hipotesi bat izan daiteke, aurre-entrenamenduan posizioaren informazioa erantsiz, arrazonamendu espaziala bezalako gaitasunak lortzeko gai izatea transformer batzuk, nahiz eta aurre-entrenamendu hori ez egon zuzenean bideratua gaitasun horiek ikastera.

Agian, VisualBERTen aurre-entrenamendu atazetan posizioaren informazioa nolabait gehitu izan balitz, arrazonamendu espazialerako gaitasuna garatuko zukeen, eta azken fase honetan desberdin funtzionatuko zukeen posizioaren informazioa era batera edo bestera pasatuz gero. Ideia hau geroago garatu eta aztertuko da eztabaida 4.5 azpiatalean, ViLT eta LxMERT transformerrekin konparaketa egitean, hauek aurre-entrenamenduan erabili

egiten baitute posizioaren informazioa. ViLT eta LxMERT jada aurkeztuak eta azalduak izan dira aurrekarien 2 atalean, eta, beraz, VisualBERTekin konparatu soilik egingo dira.

4.4 Analisia

Metrika orokorrak aztertu ostean, instantzia indibidualetan eredu desberdinek zein erantzun eman duten aztertuko da atal honetan. Horretarako, oinarrizko eredua, *rectangle* eta *grid* (16×16) ereduek eman dituzten erantzunak aztertu dira eskuz, instantzia indibidualak hartuz. Aztertu da ea zein galderatan eman dituzten erantzun desberdinak, eta ea nolabaiteko patroirik ikusi daitekeen erantzunetan. Noski, jarraian adibide batzuk bakarrik jarriko dira, dokumentua ez gehiegi luzatzeko, nahiz eta egin den analisian adibide gehiago begiratu diren, adibide adierazgarriak aukeratuko dira, ahal den heinean.

Lehenik eta behin, hiru adibide aukeratu dira, galderan posizioaren inolako erreferentziarik ez daukatenak. Alegia, kasu honetan ez da posizioaren inguruko ezer galdetzen, eta beraz, teorikoki, ez litzateke diferentziarik ikusi behar ereduen artean.

Hautatu diren hiru adibideetan, esperimentuen antzera, ez da patroirik ikusten, eta berdina gertatzen da eskuz egin den analisi sakonagoan. Lehenengo eta bigarren adibideetan, oinarrizko eredua da erantzun egokia lortzen duena, eta hirugarrenenean bat bera ere ez. Teorian, galdera hauek ez lukete desberdintasunik pairatu behar posizioaren kodeketaren arabera, galderek ez diotelako erreferentziarik egiten horri, baina hala gertatzen da.

Eta emaitza hauek kontsistenteak dira datu multzoko instantzietan zehar. Ez da lortu aurkitzea desberdintasun hauek esplika ditzakeen inolako patroirik. Batzuetan eredu batek asmatzen du galdera, eta besteetan beste batek, edo hirurek, edo batek ere ez. Ereduen entrenamenduan diferentziak sortzen dira, noski, baina diferentzia hauek ezin dira esplikatu kodeketa ezberdinak arrazoitzat hartuta.

4.9 Irudia: 1. adibidea: 'What color is the flip flop?' Kasu honetan, *rectangle* kodeketako ereduak asmatzen du erantzuna. Iturria: [4]

Posizioaren Kodeketa	Ereduaren erantzuna	Benetako erantzuna
Oinarrizko eredua	‘orange’	[‘black’, ‘red’, ‘red’, ‘red’, ‘red and blue’, ‘red’, ‘red’, ‘red’, ‘red’, ‘red’]
Rectangle	‘red’	
Grid (16×16)	‘black’	

4.3 Taula: 1. adibidea: erantzunen konparaketa

4.9 irudiaren kasuan, esaterako, objektu detektatzaileak erdiko txankleta eta haren kolorea identifikatu behar lituzke, eta *rectangle* kodeketadun eredua izan da ondo erantzun duena. *Grid* ereduak beharbada ezkerreko txankleten kolorea erantzun du, eta besteak ez dagoen kolore bat aukeratu du.

4.10 Irudia: 2. adibidea: ‘How many mice are on the desk?’ Kasu honetan, oinarrizko ereduak eman du erantzun egokia. Iturria: [4]

Posizioaren Kodeketa	Ereduaren erantzuna	Benetako erantzuna
Oinarrizko eredua	‘1’	[‘1’, ‘2’, ‘1’, ‘1’, ‘1’, ‘1’, ‘1’, ‘1’, ‘1’, ‘1’]
Rectangle	‘2’	
Grid (16×16)	‘2’	

4.4 Taula: 2. adibidea: erantzunen konparaketa

4.10 irudiaren adibidean, irudiko sagu kopurua identifikatu behar da, eta oinarrizko eredua izan da erantzun egokia eman duen bakarra. Beste bi ereduek 2 sagu eman dute erantzun bezala, nahiz eta irudian ez den bigarrenik ikusten.

4.11 Irudia: 3. adibidea: ‘What color is the Salisbury Rd. sign?’ Hiru ereduek erantzun okerra eman dute adibide honetan. Iturria: [4]

Posizioaren Kodeketa	Ereduaren erantzuna	Benetako erantzuna
Oinarrizko eredua	‘yellow’	[‘white with blue lettering’, ‘white and blue’,
Rectangle	‘black’	‘white’, ‘blue and white’, ‘white’, ‘brown’,
Grid (16×16)	‘gray’	‘white’, ‘white’, ‘not sure’, ‘white’, ‘white and blue’]

4.5 Taula: 3. adibidea: erantzunen konparaketa

4.11 irudiaren kasuan, testuaren laguntzaz lehenengo kartelaren kolorea identifikatu behar da, eta hiru ereduek erantzun okerra eman dute. Oinarrizko ereduak eta *grid* ereduak beste bi kartelen koloreak eman dituzte, eta besteak beharbada beste karteletako letren kolorea identifikatu du.

Hiru adibide horiek ikusi ondoren, jarraian beste hiru adibide erakutsiko dira, baina kasu honetan azpimultzo espazialean egongo liratekeenak. Ikusiko den bezala, hiru adibide hauek posizioaren inguruko erreferentziaren bat daukate galderan, eta beraz, instantzia

hauetan diferentziak aztertzea ere interesgarria izan daiteke.

Lehen adibidean, hiru ereduek okerreko erantzuna ematen dute (nahiz eta *grid* kodeketadunarena egoki bezala interpretatu daitekeen). Bigarrenean, hiru ereduek egoki erantzuten dute, eta badirudi gai direla ezkerreko ibilgailua zein den identifikatzeko. Hirugarrenean, berriz, hirurek oker erantzuten dute (nahiz eta *rectangle* kodeketadun sarearen erantzuna ontzat eman daitekeen, beharbada).

Alegia, berriro ere, adibide hauekin ilustratu nahi dena da, kodeketa bat ez dela bestea baino hobea. Ikusten da ereduen arteko diferentziak ezin direla zuzenean korrelazionatu posizioaren kodeketarekin. Adibidez, lehen irudiaren kasuan, zergatik erantzuten du oinarrizko ereduak 'fork', irudian sardexkarik ez bada agertzen? Izan ere, badirudi hori objektu detektatzailearen arazo bat izan daitekeela, edo testuingurua erabiliz emaitza bat ematen saiatu dela eredua.

Beste era batera esanda, ziurrenik, ereduen arteko desberdintasuna beraien entrenamenduan zehar ematen diren diferentzia txikiekin esplikatu daiteke. Posizioaren kodeketak barneko balioetan eragina izango du, noski, nahiz eta ezin den esan eragin hau positiboa edo negatiboa den. Soilik, ereduen arteko diferentziak sortzen ditu, entrenamendu prozesua apur bat aldatuz kasu bakoitzean, baina inolako eragin esplikagarririk gabe.

4.12 Irudia: 4. adibidea: 'What is to the right of the soup?' Hiru ereduek erantzun desberdinak baina desegokiak eman dituzte kasu honetan. Iturria: [4]

Posizioaren Kodeketa	Ereduaren erantzuna	Benetako erantzuna
Oinarrizko eredua	'fork'	['chopsticks', 'chopsticks', 'chopsticks',
Rectangle	'carrots'	'chopsticks', 'chopsticks', 'shrimp', 'chopsticks',
Grid (16×16)	'spoon'	'chopsticks', 'chopsticks', 'chopsticks spoon']

4.6 Taula: 4. adibidea: erantzunen konparaketa

4.12 irudiaren kasuan, oinarrizko ereduak irudian ez dagoen objektu baten erantzuna eman du. Bigarrenak irudiko okerreko objektu bat hartu du erantzun bezala, posizioaren informazioa gaizki erabiliz, eta azkenak, okerreko erantzuna aukeratu badu ere, errealitatera gehien hurbiltzen dena aukeratu du.

4.13 Irudia: 5. adibidea: 'What does the truck on the left sell?' Adibide honetan hiru ereduek erantzun egokia eman dute, baita posizioaren informaziorik ez duen ereduak ere. Iturria: [4]

Posizioaren Kodeketa	Ereduaren erantzuna	Benetako erantzuna
Oinarrizko eredua	'ice cream'	['ice cream', 'ice cream', 'ice cream',
Rectangle	'ice cream'	'ice cream', 'ice cream', 'ice cream',
Grid (16×16)	'ice cream'	'ice cream', 'ice cream', 'ice cream', 'ice cream']

4.7 Taula: 5. adibidea: erantzunen konparaketa

4.13 adibidean, hiru ereduak izan dira gai ezkerreko furgoneta identifikatzeko, eta baita furgoneta mota identifikatzeko ere. Kasu honetan ezkerra zein den identifikatzeko gai izan dira, nahiz eta badirudien posizioaren informazioa ez dela beharrezkoa izan, oinarrizko ereduak ongi identifikatu baitu furgoneta, posizioaren inguruko informaziorik izan gabe.

4.14 Irudia: 6. adibidea: ‘What is behind the giraffe?’ Adibide honetan, hiru ereduek oker erantzun dute. Iturria: [4]

Posizioaren Kodeketa	Ereduaren erantzuna	Benetako erantzuna
Oinarrizko eredua	‘fence’	[‘display’, ‘bird’, ‘ostrich’,
Rectangle	‘trees’	‘ostrich’, ‘ostrich’, ‘ostrich’,
Grid (16×16)	‘fence’	‘ostrich’, ‘ostrich’, ‘ostrich’, ‘ostrich’]

4.8 Taula: 6. adibidea: erantzunen konparaketa

4.14 irudiaren kasuan, hiru erantzunak okerrak izan dira. *Rectangle* ereduak irudiko beste objektu bat eman du emaitzatzat, eta beste biek irudian ez dagoen objektu bat.

4.5 Eztabaida

Beraz, lehen bi esperimentuetan ikusi den bezala, VisualBERTen posizio kodeketa ezberdinek ez dute diferentziarik suposatzen emaitzetan. LxMERT eta ViLTen emaitzekin konparatzen baditugu ordea, beste hainbat aspektu interesgarri ikus daitezke.

Berez, berriro VisualBERT LxMERT eta ViLT ekin konparatu nahi bagenitu, egokiena berriro balidazio partizio berdinean ebaluatu eta konparatzea litzateke. Hala ere, Huggingface transformers liburutegian ez daude hau egiteko esperimentua hasieratzeko behar diren pisuak. Dauden pisuetan entrenamendua egina dago jada, eta artikulu originaletara bagoaz, ikusten dugu entrenamendurako, entrenamendu partizioko instantziak erabiltzeaz gain, balidazio partizioko instantziak erabiltzen dituztela, gero test partizioan neurtzeko

emaitzak. Baina, guk balidaziokoan neurtu nahi ditugunez, metodologikoki ez litzateke egokia, bertako hainbat instantzia jada ezagunak zaizkielako LxMERT eta ViLTi. [1]

Horregatik, beste artikulu bateko emaitzak hartu dira konparaketa egiteko, zehazki, ViLTen artikulu [8] originalekoak. Artikuluan VQA v2.0 datu multzoan hainbat transformerrek lortzen dituzten emaitzen laburpen bat dago. Bertan aurkitzen dira, noski, ViLT, eta baita LxMERT eta VisualBERT ere. 4.9 taulan ikus daitezke VQA v2.0 datu multzoko test partizioan hiru transformerrek lortzen dituzten emaitzak. Hiru transformerrak desberdinak dira elkarren artean, eta entrenamendu prozesu desberdinak dituzte, baina, hala ere, antzeko emaitzak lortzen dituzte test partizioaren gainean.

Eredua	Asmatze-tasa (%) (VQA v2.0 test-dev)
LxMERT	72,42
ViLT	71,26
VisualBERT	70,80

4.9 Taula: LxMERT, ViLT eta VisualBERTen emaitzak, VQA v2.0 test partizioan

LxMERTek du emaitza onena, ondoren ViLTek, eta azkenik VisualBERTek, baina emaitzen arteko diferentziak oso txikiak dira, eta ereduen desberdintasunekin esplika daitezke. Izan ere, hiru transformerren arteko diferentziak ez dira posizio kodeketarenak bakarrik; bakoitzak arkitektura ezberdinak, aurre-entrenamendu ezberdinak eta entrenamendurako datu multzo ezberdinak erabiltzen ditu.

Diferentzia horiengatik, emaitza hauekin ezin da ondorioztatu informazio espaziala ulertu eta ikasteko ereduek duten gaitasuna. Hori dela eta, arrazonamendu espazialaren analisia sakonago egiten duen artikulu bat hartuko da oinarritzat, lehen esan bezala, "Visual spatial reasoning" [9]. Artikulu horretan, VSR datu multzoa aurkezten da, arrazonamendu espazialaren inguruko galderak soilik dituelarik.

VSR datu multzoa, artikuluan proposatzen den datu multzoa da, sortutako ataza berrirako. Datu multzo honen helburua, bereziki arrazonamendu espaziala neurtzea da, datu sorta orokorragoen aldean, txikiagoa izanik ere, bereziki posizioaren eta orientazioaren inguruan etiketatutako datuak dituelako.

Datu sorta honek irudi-deskribapen datu bikoteak ditu instantziatzat. Alegia, kasu honetan ez daude irudiari buruzko galderak, baina ataza galdera bati erantzutea da: bat al datoz deskribapena eta irudia? Teknikoki, beraz, ez da VQA ataza bat, baina irudiaren eta testuaren ulermenarekin bat datorren ataza bat da. Gainera, lehen esan bezala, arrazonamendu espazialaren inguruko datu sorta bat denez, deskribapen guztietan egongo da erlazio espazialen bat. Beste era batera esanda, inoiz ez da agertuko "Pertsona bat bazkaria jaten ari da" bezalako deskribapenik, guztiak espazioarekin lotuta egongo dira, "Behia pertsonaren ezkerrean dago" bezala, adibidez.

Horiek horrela, datu multzoak, COCO datu sortako 6.940 iruditatik abiatuta, 10.119 instantzia ditu. Kontuan izan behar da, berriro, instantzia bakoitzak irudi bat, deskribapen bat eta erantzun bat dituela (egia/gezurra). 4.15 irudian ikusi daitezke VSR datu multzoko bi instantzia, bakoitza bere deskribapen eta erantzunarekin.

[1] Zehazki, LxMERTek balidazio partizioko 5000 instantzia erreserbatzen ditu balidazio partizio propio bat sortzeko, eta ViLTek 1000. Ondoren, geratzen diren beste instantzia guztiak entrenamenduan erabiltzen dituzte.

Caption: *The person is ahead of the cow.*
Label: `True`.

Caption: *The cat is inside the toilet.*
Label: `False`.

4.15 Irudia: VSR datu multzoko bi instantzia. Irudi bat eta honen gaineko esaldi bat hartuta, esaldia bat datorren edo ez adierazi behar da. Gainera, esaldiak beti objektuen posizioaren inguruko erreferentziaren bat eduki beharko du. Iturria: [9]

Konparaketarekin hasiz, artikulu originalean aurkezten diren emaitzen laburpen bat da 4.10 taula. LxMERT, ViLT eta VisualBERTen emaitzak konparatzen dira, VSR datu multzoko *random split* partizioan, [2] gizakiek lortzen dituzten emaitzekin alderatuz, baita ere.

Lehenik ikusi daitekeena da gizaki emaitzatik oso urrun daudela hiru transformerrak, hurbilen dagoenera, LxMERTera, 23 puntu inguruko diferentzia baitago. Beraz, gizakientzat berez nahiko erraza dirudien ataza bat nahikoa zailagoa da transformer hauentzat. [3]

Eredua	Asmatze-tasa (%) (VSR random split)
Gizaki-emaitza	$95,4$
LxMERT	$72,5_{\pm 1,4}$
ViLT	$71,0_{\pm 0,7}$
VisualBERT	$57,4_{\pm 0,9}$

4.10 Taula: LxMERT, ViLT eta VisualBERTen emaitzak, VSR random split partizioan

Baina, batez ere, taulan ikusten dena da diferentzia handia dagoela VisualBERTen eta beste bi transformerren artean. VQA v2.0 datu multzoan asmatze-tasak oso antzekoak izan dira, eta bazirudien hiru ereduek antzeko gaitasuna zutela irudien gaineko galderei erantzuteko; hala ere, taula honetan ikus daiteke ulermen espazialean alde izugarria dagoela VisualBERTen eta beste bi ereduen artean. VisualBERTen eta LxMERTen artean 15,1 puntuko diferentzia ikus daiteke asmatze tasan.

Artikuluaren egileek ondorioztatzen dute, aurre-entrenamendutik hasita, posizioaren kodeketa esplizituak eragiten dituela diferentzia hauek. Lehen esan bezala, LxMERTek

[2] VSRko partizioen informazio gehiago artikulu originalean aurki daiteke.

[3] Kontuan izan behar da, taulako emaitzak sortzeko, hainbat hazi desberdin erabili dituztela egileek, ondoren batez bestekoa hartzeko. Asmatze-tasaren alboan ikus daitekeen zenbakia lortutako desbideratze estandarra da.

detektatutako objektuei posizioaren informazioa gehitzen die, eta ViLTek sortutako irudi zatien posizioa ere erabiltzen du sarrera bezala. Aldiz, VisualBERTek ez du posizioaren informaziorik erabiltzen aurre-entrenamenduan.

Horregatik, esan daiteke aurre-entrenamenduan posizioaren informazioa erabiltzen ikasteko gai direla transformer hauek, nahiz eta aurre-entrenamenduko atazak ez dauden zuzenean bideratuta erlazio hauek ikastera. Badirudi, LxMERT eta ViLTek zeharka ikasten dutela posizioaren informazioa erabiltzen aurre-entrenamenduan zehar, eta VQA eta VSR atazei aurre egiterako garaian erabiltzeko gai direla. Ideia hau geroago berrikusiko da ondorioen 5 atalean.

Azkenik, aipagarria da baita ere, soilik LxMERT eta ViLT hartuta ez dela diferentzia berezirik ikusten bi transformerren artean. Egia da LxMERTen asmatze tasa ViLTena baino 1,5 puntu altuagoa dela, baina desbideratze estandar handiagoa du, eta desberdintasuna ez da bereziki handia.

Gainera, diferentzia hori esplikatzen saiatzen bagara, ezin zaio zuzenean posizioaren kodeketa desberdinari esleitu diferentzia. Alegia, ereduak beraien artean oso desberdinak dira egitura, aurre-entrenamendu eta entrenamendu aldetik, eta beraz, ezin da esan LxMERTen posizioaren kodeketa ViLTena baino hobea denik VSR atazan. Atera daitekeen ondorio bakarra da, LxMERTek ViLTek baino emaitza hobea lortzen duela atazan, eta eredua pixka bat hobea dela ataza horretarako, ez posizioaren kodeketa.

KAPITULUA 5

Ondorioak eta etorkizuneko lana

Lan honetan zehar, irudia eta testua konbinatzen dituzten transformer multimodaletan irudietako objektuen posizioen kodeketak duen eragina aztertu da. Horretarako, VQA ataza eta VisualBERT transformerra oinarritzat hartu dira, zeinak ez duen posizioaren informaziorik erabiltzen inferentzian. Ondoren, posizioaren informazioa kodetzeko hainbat forma desberdin inplementatu dira (*grid encoding* eta *rectangle encoding*) eta elkarren artean konparatu dira.

Konparaketa hauek VQA v2.0 datu multzoaren gainean egin dira lehenik, eta, ondoren, arrazonamendu espaziala zehatzago neurtzeko, datu multzo honen azpimultzo espaziala proposatu da, eta bertan egin da konparaketa. Azkenik, beste bi transformer multimodal hartu dira, LxMERT eta ViLT, posizioa kodetzeko era desberdinak inplementatzen dituztenak, eta hauekin konparatu da VisualBERT, beste artikulu batzuetako emaitzei erreferentzia eginez.

Analisi honen ostean, hainbat ondorio interesgarri atera dira, posizioaren kodeketaren inguruan aspektu interesgarriak erakusten dituztenak etorkizunean egin daitezkeen lanetarako.

Lehenik eta behin, lanean atera den ondorio garrantzitsuena da VisualBERT ez dela gai posizioen *embeddingak* ikasteko. Entrenamenduko azken fasean posizioaren kodeketa desberdinak inplementatuta, VisualBERT ez da gai izan posizioaren informazio hau erabiltzeko. Hau argi ikusi da, VQA v2.0 datu multzoaren gainean ez delako diferentziarik egon oinarrizko ereduaren, *grid* eta *rectangle* kodeketen artean. Azpimultzo espazialean ere berdina gertatu da, ez da diferentziarik egon, eta gainera emaitza orokorrak okerragoak izan dira, VisualBERTek erlazio espazialak ikasteko duen zailtasuna erakutsiz.

Hori kontuan hartuz, ezin da garbi esan posizioaren zein kodeketa den onena. Horren gaineko ondorio bat atera nahi bada, lehenik eta behin posizioaren informazioa erabiltzeko gai den eredu bat eduki beharko litzateke, ondoren konparaketa egiteko, beraz, ez da ondorioztatu zein kodeketa den egokiena.

Hala ere, badirudi ondorioztatu ahal izan dela aurre-entrenamenduak duen garrantzia posizioaren *embeddingak* erabiltzen ikasteko. Ikusi da, LxMERT, ViLT eta VisualBERTek VQA atazan antzeko emaitzak lortzen badituzte ere, VSR atazan askoz emaitza hobeak

lortzen dituztela lehen biek. Posible da, VQA atazan posizioaren informazioa ez hain garrantzitsua izatea, baina ez da probablea.

Probableagoa da, aurre-entrenamenduan posizioaren *embeddingak* sartzen badira, ereduak gai direla *embedding* horiek erabiltzen ikasteko. Izan ere, VSR ataza sortu eta aztertu zuten egileek ondorioztatzen dute atazan lortzen diren emaitzen diferentzia horrela esplikatu daitekeela, LxMERT eta ViLT VisualBERTengandik hala desberdintzen baitira.

Hala ere, nahiz eta aurre-entrenamenduan posizioaren *embeddingak* sartuz badirudien emaitzak hobetzen direla, ezin da determinatu zein kodeketa den hobea, LxMERTek erabiltzen duen *rectangle encodingaren* antzeko kodeketa edo ViLTek erabiltzen duen irudia zatikatzearen kodeketa. LxMERT eta ViLTek sare arkitektura desberdinak erabiltzen dituzte, eta aurre-entrenamendu eta entrenamendu prozesuak oso desberdinak dira. Beraz, nahiz eta emaitza desberdinak lortu, ezin dira diferentzia hauen arrazoiak zuzenean posizioaren *embedding* desberdinei esleitu. Alegia, ezin da esan ereduen arteko desberdintasunak posizioen *embeddingek* sortzen dituztenik, beharbada beste faktoreek eragin ditzaketelako.

Hala ere, badirudi, bi kodeketak egokiak izan daitezkeela, baldin eta aurre entrenamenduan sartzen badira. Horiek horrela, azken aldian garatzen ari den kodeketa zatikakoa izaten ari da, bere sinpletasunagatik, azkarragoa baita objektu detektatzaile bat erabiltzea baino, eta honekiko dependentzia galtzen delako.

Azkenik, lan honetan arrazonamendu espazialean posizioaren kodeketen garrantzia ikertu bada ere, transformerren beste aspektuek ere eragina izan dezakete beharbada, eta aztergai interesgarriak izan daitezke, hala nola arkitektura berriak edo entrenamendu espezifikoagoak.

Etorkizuneko lanari begira, oso ikerketa lerro interesgarria izan daiteke hori. Aurre-entrenamenduaren garrantzia ia segurua denez, aurre-entrenamendu egokiak zein diren aztertu daiteke arrazonamendu espaziala ondo egiterako garaian. Are gehiago, lerro honi jarraituz, aurre-entrenamendu ataza berriak diseinatzea ere interesgarria izan daiteke, hobe betetzeko atazak dituen beharrak eta zailtasunak gainditzen laguntzeko.

Aurre-entrenamendu hauen eraginkortasuna ikusteko bide posible bat izan daiteke berriro VisualBERT oinarritzat hartu eta aurre-entrenamendu desberdinetan posizioaren kodeketa desberdinak inplementatzea. Hemen, alde batetik, jada VisualBERTek egiten dituen aurre-entrenamenduak egin daitezke, alegia, *masked language modeling* irudiarekin eta *sentence-image prediction.* Ondoren, entrenamendu hauetan posizioaren kodeketak integratuko lirateke, adibidez, *rectangle*, *grid* eta zatikako kodeketa, eta azkenik, *fine-tuning* fasearen ondoren, berriro VQA v2.0 eta azpimultzo espazialean lortutako emaitzak dokumentu honetan aurkeztutakoekin konparatu daitezke.

Beste aldetik, kodifikazio espezifiko bat aukeratu, eta aurre-entrenamendu desberdinak probatu daitezke, gero emaitzak berriro baldintza berdinetan neurtzeko. Horrela, aztertu daiteke zein aurre-entrenamendu den egokiena arrazonamendu espaziala hobetzeko.

Eta, noski, azterketak ez du zertan VisualBERTera mugatua egon behar. Nahiz eta lan honen aztergaia transformer hau izan den, ikusi diren beste transformerrekin, hala nola, ViLT edo LxMERTekin ere probatu daitezke aurre-entrenamendu berriak, ikusteko ea eredu desberdinetan zehar eraginkortasuna mantentzen den, eta hala bada, etorkizunean sortuko diren ereduetan integratu daitezke aurre-entrenamendu hauek, arrazonamendu espaziala hobetzeko helburuarekin.

Gainera, kodeketa eta aurre-entrenamenduak eredu desberdinetan probatzen badira, prozesu horiek erabat kopiatuz, baita ere aztertu daiteke zein eredu litzatekeen onena arrazonamendu espazialerako entrenamendua eta kodeketa kontuan hartu gabe.

Bestalde, VSR ataza eta datu multzoa oso berriak dira, kontuan hartuz 2022ko apirilaren 30ekoa dela artikulua. Horregatik, etorkizunerako hainbat ikerketa burutu daitezke lerro honi jarraiturik. Hasteko, VQA v2.0 datu multzoarekin alderatzen badugu VSR, datu multzoen tamainen artean alde oso handia dago. VSR datu multzoa handitzeak asko lagundu dezake datu multzo osatuago bat egiten. Adibidez, datu multzo berri hau oso ona izan daiteke justu lehen aipatutako aurre-entrenamenduak egiteko, ereduek erlazio espazialak ikasteko. Agian, proposatu den VQA v2.0ren azpimultzo espaziala abiapuntu egokia izan daiteke datu multzo hau egiten hasteko, ondoren, irizpide ezberdinen arabera filtratuz, eta instantziak egokituz.

Datu multzoa hobetzeaz gain, berriro ere aurre-entrenamenduen diseinuaren ideia gogoratuz, jada gaur egun duen egoera ere aprobetxatu daiteke. Adibidez, aztertu daiteke VSR datu multzoa eta ataza oinarritzat hartuta, aurre-entrenamendutzat erabiltzen bada, ea nolako eragina izango lukeen horrek ondoren, VQA atazan lortzen diren emaitzetan, bai VisualBERTekin edo baita beste ereduekin ere.

Hainbat ikerketa lerro posible izan daitezke oraintxe aurkeztu direnak, izan ere, oraindik esploratzeko asko duen arloa da VQA atazan arrazonamendu espazialarena. VSR artikulua pausuak ematen hasi den arren, nahiko artikulu berria da, eta beraz hobetzeko aukera handia dago oraindik, ikusi den bezala.

Eranskina

Azpimultzo espazialerako hitz zerrenda osoa

["touching", "behind", "in front of", "on", "under", "on top of", "at the right side of", "at the left side of", "contains", "beneath", "above", "next to", "facing", "in", "below", "inside", "far away from", "at the edge of", "left of", "beside", "facing away from", "away from", "far from", "part of", "near", "right of", "close to", "across from", "surrounding", "at the back of", "parallel to", "in the middle of", "over", "adjacent to", "off", "perpendicular to", "attached to", "by", "at the side of", "alongside", "against", "ahead of", "consists of", "toward", "within", "outside", "connected to", "opposite to", "into", "has a part", "enclosed by", "with", "beyond", "across", "down from", "detached from", "out of", "around", "at", "past", "along", "between", "down", "among", "big", "bigger", "biggest", "little", "small", "smaller", "smallest", "large", "larger", "largest", "tiny", "enormous", "huge", "gigantic", "long", "longer", "longest", "short", "shorter", "shortest", "tall", "taller", "tallest", "wide", "wider", "widest", "narrow", "narrower", "narrowest", "thick", "thicker", "thickest", "thin", "thinner", "thinnest", "skinny", "fat", "fatter", "fattest", "deep", "deeper", "deepest", "shallow", "shallower", "shallowest", "up", "bottom", "high", "low", "column", "vertical", "left", "right", "front", "back", "ahead", "sideways", "row", "horizontal", "close", "next", "far", "away", "together", "separate", "join", "apart", "middle", "center", "north", "south", "east", "west", "opposite", "reverse", "backward", "forward", "parallel", "perpendicular", "diagonal", "location", "position", "direction", "route", "path", "place", "distance", "upside down", "right side up", "upright", "turn", "flip", "rotate", "rotation", "whole", "part", "piece", "section", "segment", "portion", "fragment", "fraction", "half", "more", "less", "same", "equal", "amount", "space", "area", "where", "side", "edge", "border", "line", "before", "after"]

Bibliografia

[1] Aditya Mogadala, Marimuthu Kalimuthu, and Dietrich Klakow. Trends in integration of vision and language research: A survey of tasks, datasets, and methods. *Journal of Artificial Intelligence Research*, 71:1183–1317, 2021. Ikusi vii, 5, 6, and 9 orrialdeak.

[2] Haotian Zhang, Pengchuan Zhang, Xiaowei Hu, Yen-Chun Chen, Liunian Harold Li, Xiyang Dai, Lijuan Wang, Lu Yuan, Jenq-Neng Hwang, and Jianfeng Gao. Glipv2: Unifying localization and vision-language understanding. *arXiv preprint arXiv:2206.05836*, 2022. Ikusi vii, 5 orrialdeak.

[3] Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. From recognition to cognition: Visual commonsense reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. Ikusi vii, 6 orrialdeak.

[4] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6904–6913, 2017. Ikusi vii, viii, 2, 7, 8, 34, 35, 36, 37, 38, and 39 orrialdeak.

[5] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, AidanÑ Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. Ikusi vii, 1, and 9 orrialdeak.

[6] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*, 2019. Ikusi vii, 1, 4, 10, 13, 14, and 27 orrialdeak.

[7] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, 2019. Ikusi vii, 1, 10, and 11 orrialdeak.

[8] Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. In *International Conference on Machine Learning*, pages 5583–5594. PMLR, 2021. Ikusi vii, 1, 10, 13, 14, 15, 27, and 40 orrialdeak.

[9] Fangyu Liu, Guy Emerson, and Nigel Collier. Visual spatial reasoning. *arXiv preprint arXiv:2205.00363*, 2022. Ikusi ix, 22, 23, 40, and 41 orrialdeak.

[10] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32, 2019. Ikusi 1, 4 orrialdeak.

[11] Emanuele Bugliarello, Ryan Cotterell, Naoaki Okazaki, and Desmond Elliott. Multimodal Pretraining Unmasked: A Meta-Analysis and a Unified Framework of Vision-and-Language BERTs. *Transactions of the Association for Computational Linguistics*, 9:978–994, 09 2021. Ikusi 1 orrialdea.

[12] Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y. Ng. Multimodal deep learning. In *ICML*, pages 689–696, 2011. Ikusi 3 orrialdea.

[13] Tadas Baltrušaitis, Chaitanya Ahuja, and Louis-Philippe Morency. Multimodal machine learning: A survey and taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2):423–443, 2019. Ikusi 4 orrialdea.

[14] Alane Suhr, Mike Lewis, James Yeh, and Yoav Artzi. A corpus of natural language for visual reasoning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 217–223, 2017. Ikusi 5 orrialdea.

[15] Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. A corpus for reasoning about natural language grounded in photographs. *arXiv preprint arXiv:1811.00491*, 2018. Ikusi 5 orrialdea.

[16] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer vision*, pages 2641–2649, 2015. Ikusi 5 orrialdea.

[17] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73, 2017. Ikusi 5 orrialdea.

[18] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015. Ikusi 6 orrialdea.

[19] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, 2018. Ikusi 6 orrialdea.

[20] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433, 2015. Ikusi 7 orrialdea.

[21] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. Ikusi 10 orrialdea.

[22] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015. Ikusi 10 orrialdea.

[23] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. Ikusi 15 orrialdea.

[24] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 702–703, 2020. Ikusi 16 orrialdea.