

VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
PROGRAMŲ SISTEMŲ BAKALAURO STUDIJŲ PROGRAMA

Nuostolio funkcijų taikymo vaizdų panašume tyrimas

A study of loss functions application in image similarity

Bakalauro baigiamasis darbas

Atliko: Gabija Gakaitė
Darbo vadovas: j. asist. Boleslovas Dapkūnas
Darbo recenzentas: doc. dr. Valentas Gružas

Vilnius – 2024

Autoriaus padėka

Darbo autorius dėkoja Vilniaus universiteto Matematikos ir informatikos fakulteto Informacinių technologijų atviros prieigos centrui už suteiktus HPC išteklius šio darbo skaičiavimams atlikti.

Santrauka

Kartu su išmaniosiomis technologijomis išaugus lengvai prieinamų vaizdinių duomenų kiekiui, atsiranda poreikis gebėti rasti, atpažinti bei palyginti paveikslėlius naudojant ne tik tekstines bet ir vaizdines užklaudas. Plačiai tyrinėjamas vaizdų panašumo palyginimo uždavinių sprendimo būdas yra dirbtiniai neuroniniai tinklai. Šio tyrimo objektas – nuostolio funkcijos, esminis komponentas šių tinklų apmokymo procese. Darbo tikslas yra išnagrinėti vaizdų panašumo srityje naudojamas nuostolio funkcijas, išsiaiškinti, su kuria funkcija apmokyti modeliai geriausiai sprendžia vaizdų atpažinimo bei atrinkimo uždavinius, bei panagrinėti, kaip modelių rezultatai priklauso nuo duomenų išgavimo panaudojimo. Skirtingo gylio ResNet architektūra paremti modeliai buvo apmokyti naudojant „Google Landmarks Dataset v2“ duomenų rinkinio poaibį. Modelių vertinimui naudoti „Oxford Buildings“, „Paris Dataset“ bei „Revisited Oxford“, „Revisited Paris“ duomenų rinkiniai. Geriausias vidutinio preciziškumo vidurkis (mAP) įverčius, nenaudojant porų išgavimo mokymo metu, pasiekė CosFace nuostolių funkciją naudojant apmokyti tinklai. Porų išgavimo mokymo metu panaudojimas pagerino modelių pasiekiamus rezultatus. Pačius geriausius rezultatus pateikė modeliai, apmokyti naudojant porų išgavimą mokymo metu bei SphereFace nuostolių funkciją.

Raktiniai žodžiai: nuostolių funkcijos, vaizdų atrinkimas, vaizdų atpažinimas, duomenų išgavimas, konvoliuciniai neuroniniai tinklai

Summary

Together with new technologies came the increase in the amounts of readily available visual data, creating the need to be able to find, recognize, and compare images using not only textual but also visual queries. A widely researched method for solving such image similarity comparison problems is artificial neural networks. This study focuses on loss functions – an integral part in the training process of such networks. The aim of this work is to research the loss functions used in the field of image similarity, determine which function-trained models best solve image recognition and retrieval tasks, and analyse how model performance depends on the use of data mining. Models based on different depths of ResNet architecture neural networks were trained using a subset of the “Google Landmarks Dataset v2” (GLDv2) dataset. For model evaluation the “Oxford Buildings” and “Paris Dataset” together with “Revisited Oxford” and “Revisited Paris” datasets, were used. The best mean average precision (mAP) scores, without using online pair mining, were achieved by networks trained with the CosFace loss function. The use of online pair mining improved the models’ performance. The overall best results were obtained by models trained using online pair mining and the SphereFace loss function.

Keywords: loss functions, image retrieval, image recognition, data mining, convolutional neural networks

Turinys

ĮVADAS	7
1. KONVOLIUCINIAI NEURONINIAI TINKLAI	9
1.1. GeM sutelkimo sluoksnis	9
1.2. Žinių pernešimo bei modelio adaptavimo metodai.....	10
1.3. Požymių vektorių vertinimo metodai	11
1.3.1. Euklido atstumas.....	11
1.3.2. Kosinuso panašumas	11
1.3.3. HOW metodas	12
2. NUOSTOLIŲ FUNKCIJOS	13
2.1. Kontrasto nuostolio funkcija	13
2.2. SupCon nuostolio funkcija	13
2.3. Trejetų nuostolio funkcija	14
2.4. A-Softmax nuostolio funkcija.....	15
2.5. CosFace nuostolio funkcija	16
2.6. ArcFace nuostolio funkcija	17
2.6.1. SubcenterArcFace nuostolio funkcija	18
2.7. Nuostolio funkcijų teorinis palyginimas	20
2.8. Trejetų bei porų išgavimo strategijos	21
3. EKSPERIMENTAS	22
3.1. Tyrimo eiga	22
3.1.1. Naudota modelių architektūra.....	23
3.1.2. Naudotos priemonės	23
3.2. Duomenys	24
3.2.1. „Google Landmarks Dataset v2“	24
3.2.2. „Oxford Buildings“ ir „Paris Dataset“	24
3.2.3. „Revisited Oxford“ ir „Revisited Paris“.....	25
3.3. Eksperimente naudoti duomenų rinkinių poaibiai	25
3.3.1. Duomenų augmentacijos	26
3.4. Naudotos trejetų ir porų išgavimo strategijos.....	27
3.4.1. Trejetų išgavimo strategija.....	27
3.4.2. Porų išgavimo strategija	27
3.5. Modelių mokymo procesas.....	28
3.5.1. Hiperparametrų parinkimas	28
3.5.1.1. Epochų skaičiaus parinkimas.....	28
3.5.1.2. Mokymo žingsnio bei jo planuotojo parinkimas	29
3.5.1.3. Duomenų išgavimo naudojimas.....	29
3.6. Vertinimo metrikos	30
3.7. Eksperimento rezultatai	32
3.7.1. Nuostolių funkcijų pritaikymo palyginimas – mAP įvertis.....	32
3.7.2. Nuostolių funkcijų pritaikymo palyginimas - mP@K įvertis.....	34
3.7.3. Duomenų atrinkimo pritaikymo tyrimas	35
REZULTATAI	36
IŠVADOS	37
ŠALTINIAI	38

SĄVOKŲ APIBRĖŽIMAI	40
PRIEDAI	41
1 priedas. Nuostolio funkcijų pritaikymo palyginimas – rparis6k mP@K rezultatai	41
2 priedas. Nuostolio funkcijų pritaikymo palyginimas – gldv2_tiny mAP, MRR bei P@1 rezultatai	42
3 priedas. Duomenų išgavimo pritaikymas – roxford5k bei rparis6k mAP rezultatai	43
4 priedas. Duomenų išgavimo pritaikymas – roxford5k bei rparis6k mP@K rezultatai ...	44

Įvadas

Vaizdų panašumo problema susijusi su tokiais uždaviniais kaip veidų atpažinimas, klasifikavimas, dublikatų paieška bei panašių vaizdų atrinkimas [WAC⁺20]. Pagrindinis klausimas skamba paprastai – kaip įvertinti semantinę dviejų vaizdų panašumą? Tačiau problemos sprendimas ne toks paprastas – naivus pikselių reikšmių palyginimas neveiksmingas, kadangi dvi to paties objekto nuotraukų pikselių reikšmės gali būti visiškai skirtingos (1 pav.). Vienas galimų sprendimų yra paveikslėlius pakeisti jų reprezentacijomis, perteikiančiomis kontekstinę informaciją. Šios reprezentacijos vadinamos esminių požymių vektoriais, jos ir naudojamos semantiniame vaizdų panašumo įvertinime.



(a) Originalus paveikslėlis

(b) Tamsesnė versija

1 pav. Du panašūs, bet nevienodi to paties pastato paveikslėliai.¹

Esminių požymių išgavimo uždavinį labai gerai atlieka konvoliuciniai neuroniniai tinklai (KNT). Šių tinklų mokymo procese svarbus vaidmuo tenka nuostolių funkcijoms – jos nurodo, kiek modelio pateikiamas rezultatas skiriasi nuo siekiamo. Tinklo parametrai koreguojami siekiant gauti kuo mažesnes nuostolio funkcijos reikšmes, taip apmokant tinklus spręsti specifinius uždavinius. Nuostolio funkcijos parinkimas nulemia modelio apmokymo trukmę, jautrumą triukšmingiems duomenims, mokymui reikalingą duomenų formatą bei pasiekiamus rezultatus. Gausu straipsnių, pristatančių naujas nuostolio funkcijas ([DGX⁺19; HFW⁺20; KTW⁺20; LWY⁺17; SKP15; WWZ⁺18] ir kt.), tačiau juose nepristatomas išsamus nuostolio funkcijų idėjų bei jomis apmokyty modelių rezultatų palyginimas. Su vaizdų panašumu susijusiose funkcijose dažnai svarbus informatyvių pavyzdžių parinkimas taip pat ne visada įtraukiamas į palyginimus. Šiuo tyrimu siekiama užpildyti šias spragas.

Tikslas

Išnagrinėti ir palyginti pasirinktas nuostolių funkcijas, bei jų pritaikymo panašių vaizdų paieškos, specifiskai žymių vietų atpažinimo bei vaizdų atrinkimo uždaviniams spręsti. Įvertinti duomenų atrinkimo svarbą nuostolio funkcijų pasiekiamiems rezultatams.

¹Autoriaus asmeninio archyvo nuotraukos.

Uždaviniai

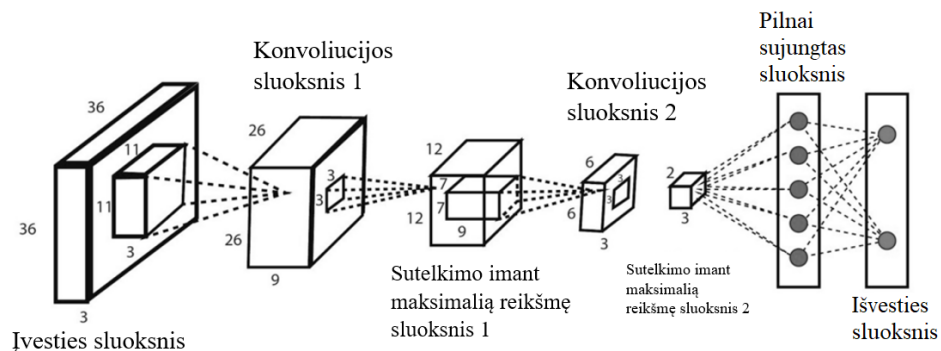
- Remiantis moksline literatūra pasirinkti iki 10-ties nuostolio funkcijų tyrimui.
- Apibrėžti eksperimentą – pasirinkti konvoliucinių neuroninių tinklų architektūrą, duomenų aibes, rezultatų vertinimo metrikas bei metodus.
- Atlikti bandymus reikalingus palyginti skirtingas nuostolio funkcijas bei įvertinti duomenų išgavimo panaudojimo poveikį, aprašyti gautus rezultatus ir išvadas.

Metodai

- Literatūros apžvalga: mokslinės literatūros apie nuostolio funkcijas, modelių architektūras, duomenų rinkinius, modelių vertinimo metrikas ir metodus paieška ir analizė.
- Empirinio eksperimento atlikimas: konvoliucinio neuroninio tinklo apmokymas naudojant skirtingas nuostolio funkcijas. Modelių įvertinimas bei rezultatų palyginimas.

1. Konvoliuciniai neuroniniai tinklai

Konvoliuciniai neuroniniai tinklai, dažnai trumpinami KNT (angl. *Convolutional Neural Networks - CNN*) yra dirbtiniai neuroniniai tinklai paremti konvoliucijos operacijomis ir naudojami kompiuterinės regos uždaviniams spręsti. Panaudojant konvoliucijos operacijas iš vaizdų išgaunami esminių požymių vektoriai (angl. *feature vectors*), kurie perneša paveikslėlyje esantį kontekstą neprisiršant prie konkrečių pikselių reikšmių. Konvoliucinis neuroninis tinklas paprastai apmokomas optimizuojant konvoliucinių filtrų reikšmes, dar vadinamas svoriais.



2 pav. Tipinė konvoliucinio neuroninio tinklo architektūra [Sar21].

Klasikinio KNT architektūrą sudaro keli pagrindiniai elementai: konvoliucijos operacijų sluoksniai (angl. *convolutional layer*), sekami aktyvacijos funkcijų (angl. *activation functions*) bei sutelkimo sluoksnių (angl. *pooling layers*). Tinklo struktūrą įprastai užbaigia pilnai sujungtų sąryšių (angl. *fully-connected layer*) sluoksniai, po kurių seka tinklo išvestis (angl. *output*).

Pasirinkta modelio architektūra turi įtakos modelio tikslumui (gražinamų rezultatų tinkamumui iškeltam uždaviniui spręsti) ir reikalingų duomenų kiekiui – įprastu atveju, gilesnis tinklas pateikia tikslesnes aproksimacijas tačiau didėjant sluoksnių skaičiui auga ir parenkamų tinklo svorių skaičius. Naudojant per mažą duomenų kiekį dideliu skaičiui parametru parinkti gali įvykti modelio persimokymas (angl. *overfitting*) – kai modelis per gerai prisitaiko prie mokymo duomenų, bet nesugeba spręsti keliamo uždavinio su naujais duomenimis.

Kiti modelio mokymą veikiančys elementai: nuostolio funkcijos bei optimizatoriai, jie tiesiogiai kontroliuoja mokymo procesą. Giliojo mokymo kontekste optimizatoriai yra funkcijos, koreguojančios svorių reikšmes, remiantis nuostolių funkcijų pateikiamais rezultatais. Naudojant gradientinio optimizavimo metodus reikšmės keičiamos siekiant gauti pačias mažiausias nuostolio funkcijos reikšmes (siekiama funkcijos minimumo).

1.1. GeM sutelkimo sluoksnis

Pagrindinė sutelkimo sluoksnių paskirtis yra sumažinti pateikto įvesties tenzoriaus dimensiją apibendrinant jo reikšmes. Paprastai tai būna vidurkio ar didžiosios reikšmės sutelkimo funkcijos taikomos apibendrinti požymių žemėlapiu regioną į vieną skaičių. Nors prarandama dalis informa-

cijos, toks reikšmių apibendrinimas sumažina reikalingų apmokyti parametrų skaičių, modelio persimokymo tikimybę bei padeda pagreitinti skaičiavimus.

Apibendrintas vidurkio sutelkimo (angl. *Generalized mean pooling - GeM pooling*) sluoksnis (čia ir toliau trumpinama GeM) pristatytas [RTC18] straipsnyje.

Sluoksnio funkcijos formulė:

$$\mathbf{f}^{(g)} = \left[f_1^{(g)}, \dots, f_k^{(g)}, \dots, f_K^{(g)} \right]^T$$

$$\mathbf{f}_k^{(g)} = \left(\frac{1}{|X_k|} \sum_{x \in X_k} x^p \right)^{\frac{1}{p}} \quad (1)$$

(1) formulėje:

- $\mathbf{f}^{(g)}$ - k dimensijų požymių vektorius.
- $\mathbf{f}_k^{(g)}$ - k-tosios požymių vektoriaus dimensijos reikšmė.
- x - reikšmės einamosios aktyvacijos kanale.
- $|X_k|$ - aktyvacijų požymių žemėlapiui k rinkinys.
- p - išmokstamas parametras, bendras visoms dimensijoms.

GeM sutelkimo sluoksnis yra sutelkimo vidurkinant ($p = 1$) bei sutelkimo imant maksimalią reikšmę ($p \rightarrow \infty$) sluoksnių apibendrinimas. GeM veikia reikšmes tarsi kontrasto išryškimas – labiau paveikiamos didžiausios ir mažiausios reikšmės, taip išsaugoma daugiau kontekstinės informacijos.

1.2. Žinių pernešimo bei modelio adaptavimo metodai

Žinių pernešimo (angl. *transfer learning*) metodas leidžia pritaikyti jau apmokytą neuroninį tinklą spręsti kitus uždavinius. Idėja yra paimti jau apmokyto KNT svorius, pakeisti paskutinį pilnai sujungtą sluoksnį nauju ir tęsti mokymo procesą, naudojant užduočiai specifinius duomenis ar kitas nuostolių funkcijas. Modelio adaptavimo (angl. *fine-tuning*) metodas padeda gauti geresnius rezultatus, vykdant ne tik naujai pridėto pilnai sujungto sluoksnio, bet ir visų kitų modelio sluoksnio parametrų koregavimą.

Paprastai pasirenkamas gerus klasifikavimo rezultatus pasiekiantis, gilus KNT, apmokytas su dideliais ir daug žymių turinčiais duomenų rinkiniais. Labai dažnai² yra naudojami ResNet (angl. *residual neural networks*) šeimos KNT.

Jie naudojami dėl kelių priežasčių:

- Gilūs ResNet tinklai pasiekia gerus klasifikavimo rezultatus – pavyzdžiui, ImageNet duomenų rinkinio validavimo poaibyje ResNet-152 pasiekia 80,2% tikslumą [HZR⁺16].
- Yra galimybė pasirinkti skirtingo gylio architektūrą. Nuo aštuoniolikos sluoksnių ResNet-18, kiek gilesnio ResNet-50 iki gilaus ResNet-151 - 151 sluoksnių neuroninio tinklo.

²PapersWithCode tinklalapyje matyti, kad ResNet tinklai minimi daugiau nei 2-juose tūkstančiuose straipsnių: <https://paperswithcode.com/method/resnet>

- Visų [HZR⁺16] straipsnyje minimų gilių tinklų svoriai yra viešai prieinami. Juos galima atsisiųsti iš PyTorch dokumentacijos tinklalapio.³

Šis metodas taikomas srityse, kuriose nėra galimybės gauti daug duomenų ir patys duomenys yra pakankamai panašūs (pavyzdžiui skirtingoms užduotims skirti paveikslėlių duomenų rinkiniai). Tai padeda sutaupyti mokymui reikalingų resursų kiekį, greičiau atlikti eksperimentus.

1.3. Požymių vektorių vertinimo metodai

Sprendžiant kompiuterinės vizijos uždavinius didelė svarba tenka KNT tipo modelių grąžinamoms išmoktoms paveikslėlių reprezentacijoms – požymių vektoriams. Vaizdų panašumo uždaviniuose esminiai yra šių požymių vektorių palyginimo metodai. Žemiau pateikiamos formulės bei metodai skirti įvertinti vektorių panašumą ir taikomi apmokant modelius spręsti su vaizdų panašumu susijusius uždavinius.

1.3.1. Euklido atstumas

Vertinant vektorių panašumą atsižvelgiama į dvi pagrindines vektorių savybes – jų kryptį bei didumą. Gerai žinomas metodas leidžiantis įvertinti dviejų vektorių panašumą remiantis jų didumu yra Euklido atstumo tarp jų paskaičiavimas. Vektoriai, tarp kurių Euklido atstumas nedidelis, yra arti vienas kito vektorinėje erdvėje. Tai žinant galima teigti, kad vektoriai „panašūs“.

Euklido atstumo (dar vadinamo antra norma, L2 norma) formulė:

$$d(p,q) = \sqrt{\sum_{i=1}^n (q_i - p_i)^2} \quad (2)$$

Čia p ir q – požymių vektoriai, gauti modeliui pateikus vaizdus. Jų dimensija priklauso nuo paskutinio modelio sluoksnio parametrų.

1.3.2. Kosinuso panašumas

Kiek kitoks metodas įvertinti vektorių panašumą yra kosinuso panašumo tarp vektorių įvertinimas. Šis metodas paremtas kampo tarp vektorių kosinuso įverčiu. Du vektoriai turintys didelį kosinuso panašumą vektorinėje erdvėje yra nukreipti bendra kryptimi nuo pradžios taško, vadinasi ir kampas tarp jų yra palyginus nedidelis.

(3) formulėje pateikta alternatyva Euklido atstumui - kosinuso vektorių panašumo formulė:

$$Sim(x,y) = \cos(\theta) = \frac{x \cdot y}{\|x\| \|y\|} \quad (3)$$

Čia $\|x\| = \sqrt{x_1^2 + x_2^2 + \dots + x_n^2}$ - Euklido norma vektoriui x . Kosinuso panašumas tarp vektorių nurodo, kiek vektorių kryptys sutampa. Pavyzdžiui, jei vektoriai statmeni vienas kitam, (3) funkcijos reikšmė lygi 0, jei vektoriai lygiagretūs ir tos pačios krypties, tai (3) formulės reikšmė

³<https://pytorch.org/vision/main/models/resnet.html>

lygi 1. Šiame tyrime kosinuso panašumas aktualus, kadangi jis naudojamas toliau aptariamose nuostolių funkcijose, taip pat jį galima panaudoti vertinant apmokyto neuroninio tinklo rezultatus, tiesiogiai palyginant dviejų paveikslėlių požymių vektorius.

1.3.3. HOW metodas

„HOW“ – tai metodas specifiškai pritaikytas objektų identifikavimo uždaviniui spręsti, pagrįstas lokalių požymių (požymių gautų iš nedidelio paveikslėlio regiono) vektorių agregavimu. Metodas susideda iš kelių pagrindinių dalių:

1. Išmokstama ištraukti lokalius požymius. Tai atliekama apmokant KNT tipo neuroninį tinklą. Tinklas apmokamas prižiūrimo mokymo metodu (naudojami sužymėti duomenys).
2. Gauti lokalūs požymiai sugrupuojami naudojant k-means ar panašų klasterizavimo metodą. Kiekviena lokalių požymių grupė tampa vizualiu žodžiu, atitinkančiu tam tikrą šablono paveikslėlyje.
3. Taip sudaroma paveikslėlių reprezentuojanti tokių vizualių žodžių histograma, rodanti kaip dažnai šie žodžiai pasikartoja paveikslėlyje.
4. Naudojantis ASMK (angl. *Approximate Subspace Matching Kernel*) metodu galima palyginti šias histogramas ir taip įvertinti dviejų paveikslėlių panašumą [DTP⁺21].

2. Nuostolių funkcijos

Šiame skyriuje pateikiamos vaizdų panašumo vertinimo srityje naudojamų nuostolio funkcijų pagrindinės idėjos, jų formulės bei paaiškinimai. Funkcijos pasirinktos remiantis [WMZ⁺20].

2.1. Kontrasto nuostolio funkcija

Kontrasto nuostolio funkcija (angl. *Contrastive Loss*) ima dviejų paveikslėlių požymių vektorius ir, jeigu jie panašūs, juos vektorinėje erdvėje siekia suartinti, o jei nepanašūs – atitolinti. Čia panašumui tarp vektorių įvertinti naudojamas Euklido atstumas. Paprastai su Kontrasto nuostolių funkcija naudojama neprižiūrimam mokymui, tokiu atveju teigiamas (angl. *positive*) pavyzdys yra augmentuotas atraminis vaizdas (angl. *anchor*), neigiamas (angl. *negative*) pavyzdys yra bet kuris kitas einamojo duomenų rinkinio (angl. *batch*) vaizdas. (4) formulė – kontrasto nuostolio funkcijos lygtis [KTW⁺20]:

$$\mathcal{L}_{Self-contrastive} = - \sum_{i \in I} \log \frac{\exp(z_i \cdot z_{j(i)} / \tau)}{\sum_{a \in A(i)} \exp(z_i \cdot z_a / \tau)} \quad (4)$$

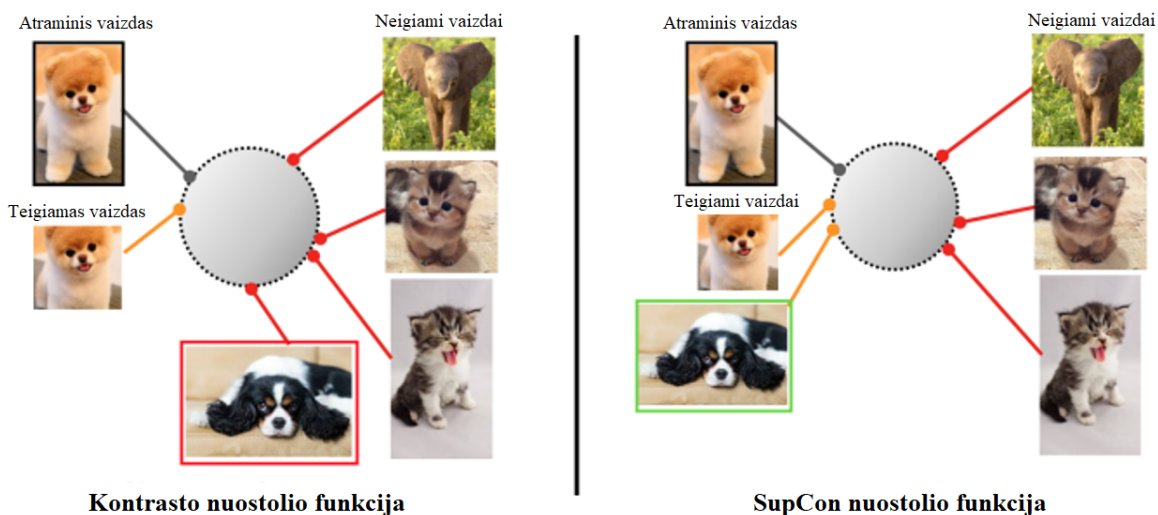
Žymėjimai:

- I - $2N$ dydžio paveikslėlių porų rinkinys naudojamas einamosios iteracijos metu. (Viename dydžio N duomenų rinkinyje susidaro $2N$ porų).
- z_i - atraminio vaizdo požymių vektorius.
- $z_{j(i)}$ - teigiamo pavyzdžio požymių vektorius.
- z_a - $2(N - 1)$ neigiamų pavyzdžių požymių vektoriai.
- τ - skaliarinis temperatūros parametras.

Taip pat $A(i) \equiv I \setminus \{i\}$ - neigiamas pavyzdys negali sutapti su teigiamu. Čia ir kitose kontrasto nuostolio funkcijos variantuose naudojamas τ parametras padeda kontroliuoti tikimybinį pasiskirstymą tarp žymių ir yra svarbus prižiūrimo mokymo atveju [WXY⁺18].

2.2. SupCon nuostolio funkcija

SupCon funkcijos pilnas pavadinimas – prižiūrima kontrasto nuostolių funkcija (angl. *Supervised Contrastive loss*). SupCon yra prižiūrimam mokymui pritaikytas Kontrasto nuostolių funkcijos variantas. Kontrasto nuostolio funkcija (4 formulė) apibendrinama leidžiant panaudoti daugiau nei vieną teigiamą porą, taip išnaudojant turimą žymių informaciją [KTW⁺20]



3 pav. Kontrasto nuostolio funkcijos ir SupCon nuostolio funkcijų palyginimas. Kontrasto nuostolio funkcijos atveju imamas vienas teigiamas pavyzdys (įprastai tai būna pasirinkto atraminio vaizdo augmentacija), o SupCon funkcija kaip teigiamus pavyzdžius paima visus tai pačiai klasei priklausančius paveikslėlius [KTW⁺20].

(5) formulė yra viena iš dviejų straipsnyje pateiktų SupCon nuostolio funkcijos variantų:

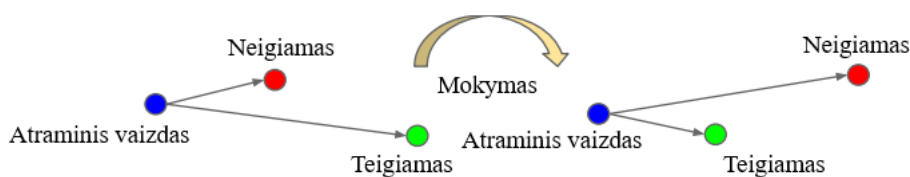
$$\mathcal{L}_{out}^{sup} = \sum_{i \in I} \mathcal{L}_{out,i}^{sup} = \sum_{i \in I} \frac{-1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp(z_i \cdot z_p / \tau)}{\sum_{a \in A(i)} \exp(z_i \cdot z_a / \tau)} \quad (5)$$

Žymėjimai:

- $P(i)$ - aibė su teigiamų pavyzdžių indeksais iš einamojo duomenų rinkinio.

Šiuo atveju teigiamų pavyzdžių sudėtis atliekama už logaritmo funkcijos ribų (todėl ir prie pavadinimo parašyta *out*). Straipsnyje pateikiamas ir kitas šios nuostolių funkcijos variantas, šis pasirinktas dėl geresnių⁴ klasifikavimo rezultatų. Naudojant daugiau nei vieną neigiamą pavyzdį, SupCon funkcija tampa ekvivalenti N-pairs nuostolių funkcijai iš [Soh16] straipsnio. Toliau nagrinėjama Trejetų nuostolio funkcija yra atskiras Kontrasto nuostolio funkcijos variantas, kai imamas vienas teigiamas ir vienas neigiamas pavyzdys [KTW⁺20].

2.3. Trejetų nuostolio funkcija



4 pav. Trejetų nuostolio funkcijos iliustracija.[SKP15]

⁴78,7% Top-1 tikslumas su (5) funkcija ir 67,4% tikslumas su alternatyvia versija [KTW⁺20].

Trejetų nuostolio funkcija naudojama siekiant apmokyti KNT iš panašių paveikslėlių gauti vektorinėje erdvėje arti vienas kito esančius požymių vektorius. Funkcijai paveikslėliai pateikiami trejetais – atraminis vaizdas, teigiamas bei neigiamas pavyzdžiai. Atraminis vaizdas ir teigiamas pavyzdys yra panašūs – tai to paties objekto nuotraukos (pavyzdžiui dvi skirtingos to paties žmogaus nuotraukos). Neigiamas pavyzdys – kito žmogaus nuotrauka [SKP15]. Kaip matyti iš 4 paveikslėlio, tikslas yra teigiamo pavyzdžio požymių vektorių vektorinėje erdvėje priartinti prie atraminio vaizdo požymių vektoriaus, o neigiamo pavyzdžio požymių vektorių – atitolinti. Atstumui tarp vektorių įvertinti naudojamas Euklido atstumas.

Supaprastinta trejetų nuostolio funkcijos formulė:

$$L_{triplet} = [d_{ap} - d_{an} + \alpha]_+ \quad (6)$$

Žymėjimai:

- d_{ap} - Euklido atstumas tarp atraminio vaizdo ir teigiamo pavyzdžio
- d_{an} - Euklido atstumas tarp atraminio vaizdo ir neigiamo pavyzdžio
- α - skirtumas (angl. *margin*), nedidelis teigiamas skaičius
- $[...]_+$ - grąžinama teigiama viduje esančio reiškinio reikšmė, arba 0 kitu atveju.

2.4. A-Softmax nuostolio funkcija

A-Softmax nuostolių funkcija (pilnas pavadinimas anglų kalba (angl. *Angular Softmax*), dažnai sutinkama SphereFace pavadinimu) atspindi kitokį požiūrį į vektorių panašumo palyginimą, nei anksčiau pristatytos SupCon bei Trejetų nuostolių funkcijos. Šiuo atveju išskirti skirtingoms klasėms priklausančių pavyzdžių požymių vektorius naudojamas ne Euklido, bet kampinis skirtumas tarp vektorių. Požymių vektoriai juos normalizuojant suprojektuojami į hipersferos paviršius (daugdarą), tada optimizuojamas kampas tarp jų. Ši nuostolių funkcija, pristatyta [LWY⁺17] straipsnyje, buvo sukurta modifikuojant Softmax nuostolių funkciją⁵ ir skirta spręsti veidų atpažinimo uždavinį. Toliau pristatoma originali Softmax funkcija bei jai atliktos modifikacijos, bei galutinė A-Softmax nuostolių funkcija, kartu su žymėjimais bei paaiškinimais:

$$\mathcal{L}_{Softmax} = -\log \left(\frac{e^{W_{y_i}^T x_i + b_{y_i}}}{\sum_{j=1}^N e^{W_j^T x_i + b_j}} \right) \quad (7)$$

$$\mathcal{L}_{Pakeista} = -\log \left(\frac{e^{\|x_i\| \cos(\theta_{y_i, i})}}{\sum_j^N e^{\|x_i\| \cos(\theta_{j, i})}} \right) \quad (8)$$

$$\mathcal{L}_{A-Softmax} = -\log \left(\frac{e^{\|x_i\| \cos(m \cdot \theta_{y_i, i})}}{e^{\|x_i\| \cos(m \cdot \theta_{y_i, i})} + \sum_{j \neq y_i}^N e^{\|x_i\| \cos(\theta_{j, i})}} \right) \quad (9)$$

Žymėjimai:

⁵Nors Softmax iš tiesų yra aktyvacijos funkcija, mokslinėje literatūroje Softmax nuostolių funkcijos pavadinimas naudojamas kaip kategorinės kryžminės entropijos (angl. *Categorical Cross-Entropy Loss*) pritaikomos iškart po aktyvacijos funkcijos, sinonimas.

- N - klasių skaičius mokymo rinkinyje.
- $W_{y_i}^T x_i + b_{y_i}$ - KNT pilnai sujungto sluoksnio išvesties, šiuo atveju klasių tikimybių vektoriaus, formulė.
- $x_i \in \mathbb{R}^d$ - i -tojo pavyzdžio požymių vektorius, atitinkantis y_i -tają žymę. (d - požymių vektoriaus dimensija).
- $W_j \in \mathbb{R}^d$ - j -toji svorių matricos $W \in \mathbb{R}^{d \times N}$ eilutė.
- b_{y_i} b_j - poslinkiai.
- θ_j - kampas tarp svorių vektoriaus W_j ir požymių vektoriaus x_i .
- θ_{y_i} - kampas tarp tikrosios žymės svorių vektoriaus W_{y_i} ir požymių vektoriaus x_i .
- m - sveikasis skaičius, kampinio atstumo parametras.

Paiškinimai:

1. (8) formulė gaunama iš (7) formulės, matricine forma užrašytą skaliarinę vektorių sandaugą $W_{y_i}^T x_i + b_{y_i}$ perrašant vektorių ilgių ir kampo tarp jų kosinuso sandauga - $\|W_{y_i}\| \|x_i\| \cos(\theta_{y_i,i}) + b_{y_i}$. Poslinkiai (b_{y_i} ir b_j) prilyginami 0, svorių vektoriai (W_{y_i} W_j) normalizuojami ir prilyginami 1.
2. Norint griežtesnio atskyrimo tarp skirtingoms klasėms priklausančių pavyzdžių požymių vektorių pridamas kampinio atstumo parametras m ($m \geq 1$). Taip norint priskirti požymių vektorių vienai klasei, kampo tarp požymio ir tos klasės svorių vektorių kosinusas turi neviršyti numatyto atstumo m (svoriai tada optimizuojami taip, kad būtų gaunamos mažesnės kampų tarp tai pačiai klasei priklausančių vektorių reikšmės).

Kadangi kosinusas yra periodinė funkcija prisideda papildomi parametrai kontroliuojantys leidžiamas kampo $\theta_{y_i,i}$ reikšmes ((9) formulėje $\theta_{y_i,i} \in [0, \frac{\pi}{m}]$). Norint, kad funkciją būtų galima naudoti su KNT, [LWY⁺17] autoriai kosinusą $\cos(\theta_{y_i,i})$ pakeičia funkcija, kuri apriboja galimas kampo reikšmes ir susieja jas su parametru m . Nauja funkcija - $\psi(\theta_{y_i,i}) = (-1)^k \cdot \cos(m \cdot \theta_{y_i,i} - 2 \cdot k)$, kur $\theta_{y_i,i} \in [\frac{k \cdot \pi}{m}, \frac{(k+1) \cdot \pi}{m}]$ ir $k \in [0, m - 1]$, todėl ir $m \geq 1$. Dėl paprastumo čia pateikiamos supaprastinta A-Softmax formulės versija.

2.5. CosFace nuostolio funkcija

CosFace nuostolio funkcija, H. Wang et. al. pristatyta [WWZ⁺18] straipsnyje didelio kosinuso skirtumo nuostolio funkcijos pavadinimu (angl. *Large Margine Cosine loss*), yra dar viena Softmax nuostolių funkcijos modifikacija. CosFace nuostolių funkcijoje atstumo parametru kontroliuojamas ne kampas tarp požymių vektorių, bet to kampo kosinusinio panašumo reikšmė (kampo tarp vektorių kosinuso didumas). Toliau pateikiamos formulės ir atlikti pakeitimai bei paaiškinimai:

$$\mathcal{L}_{Norm-Softmax} = -\log \left(\frac{e^{s \cos(\theta_{y_i})}}{e^{s \cos(\theta_{y_i})} + \sum_{j=1, j \neq y_i}^N e^{s \cos(\theta_j)}} \right) \quad (10)$$

$$\mathcal{L}_{CosFace} = \frac{1}{N} \sum_i -\log \left(\frac{e^{s(\cos(\theta_{y_i,i})-m)}}{e^{s(\cos(\theta_{y_i,i})-m)} + \sum_{j \neq y_i}^N e^{s \cos(\theta_{j,i})}} \right) \quad (11)$$

Žymėjimai:

- N - klasių skaičius mokymo rinkinyje.
- θ_j - kampas tarp svorių vektoriaus W_j ir požymių vektoriaus x_i .
- θ_{y_i} - kampas tarp tikrosios žymės svorių vektoriaus W_{y_i} ir požymių vektoriaus x_i .
- s - hipersferos, ant kurios paviršiaus nustumiami požymių vektoriai spindulys. Kitaip tariant - mastelio koeficientas (angl. *scaling factor*).

Paaiškinimai:

1. (10) formulė gaunama keičiant prie A-Softmax nuostolių funkcijos jau paminėtą originalią Softmax funkciją ((7) formulė). Lygybė $W_j^T x_i = \|W_j\| \|x_i\| \cos(\theta_j)$ gaunama taip pat kaip ir A-Softmax atveju, perrašant vektorių skaliarinę sandaugą. Poslinkį prilyginant 0 ($b_{y_i} = 0$), svorių vektorius normalizuojant ir prilyginant jį 1 ($\|W_j\| = 1$). Tačiau šiuo požymių vektorius x_i yra fiksuojamas normalizuojant naudojant Euklido normą, bei padauginant iš mastelio konstantos s (gaunama $\|x_i\| = s$).
2. Atlikus pakeitimus, vektoriaus priskyrimas klasei priklauso nuo kampo tarp požymių vektoriaus bei svorių vektoriaus reikšmės. Išmokti požymiai yra projektuojami ant s spindulį turinčios hipersferos paviršiaus, tačiau Norm-Softmax funkcija nepakankamai diskriminuoja vektorius kadangi svarba teikiama tik teisingam vektorių klasifikavimui. Tikslas yra maksimizuoti $\cos(\theta_{y_i})$ (kuo jis didesnis, tuo kampas tarp požymių vektoriaus ir tikrosios klasės svorių vektoriaus mažesnis) ir minimizuoti $\cos(\theta_j)$
3. Galutinė formulė gaunama įtraukiant kosinuso panašumo tarp vektorių atstumo parametą m . Tokiu būdu modelis mokymo metu optimizuojamas gražinti požymių vektorius, kurių kosinuso panašumas su teisingos klasės svorių vektoriais ne didesnis už $\cos(\theta_{y_i,i}) - m$. Taip nustatoma griežtesnė priskyrimo atitinkamai klasei riba ir sukuriamas atstumas tarp skirtingoms žymėms priklausančių požymių vektorių.

2.6. ArcFace nuostolio funkcija

Pilnas ArcFace nuostolio funkcijos pavadinimas – pridėtinio kampinio skirtumo nuostolio funkcija (angl. *Additive Angular Margin Loss*). Jos idėja yra optimizuoti geodezinio atstumo skirtumą (angl. *geodesic distance margin*) tarp skirtingų klasių požymių vektorių, suprojektuotų ant hipersferos paviršiaus. Geodezinis atstumas yra trumpiausias atstumas tarp dviejų taškų, esančių ant lenkto paviršiaus. Pavyzdžiui – ilgis arkos, jungiančius du sferos paviršiaus taškus. Iš čia kyla ArcFace funkcijos pavadinimo trumpinys (angl. *arc* – arka). ArcFace nuostolių funkcija, kaip ir CosFace nuostolių funkcija, gaunama modifikuojant jau pristatytos Norm-Softmax nuostolių funkcijos formulę (10). Toliau pateikiama galutinė ArcFace funkcijos formulė, kartu su žymėjimais ir atliktų modifikacijų paaiškinimais:

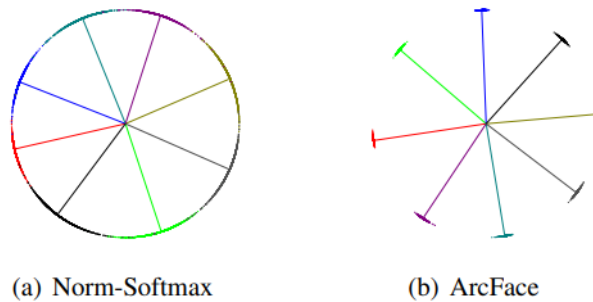
$$\mathcal{L}_{ArcFace} = -\log \left(\frac{e^{s \cos(\theta_{y_i} + m)}}{e^{s \cos(\theta_{y_i} + m)} + \sum_{j=1, j \neq y_i}^N e^{s \cos(\theta_j)}} \right) \quad (12)$$

Žymėjimai:

- N - klasių skaičius mokymo rinkinyje.
- θ_j - kampas tarp svorių vektoriaus W_j ir požymių vektoriaus x_i .
- θ_{y_i} - kampas tarp tikrosios žymės svorių vektoriaus W_{y_i} ir požymių vektoriaus x_i .
- s - mastelio koeficientas.

Paiškinimai:

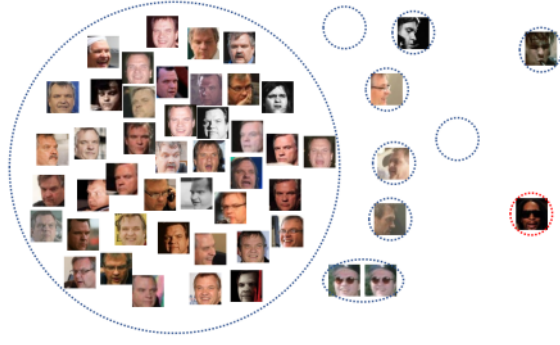
1. Esminis pakeitimas iš (10) į (12) formulę: prie θ_{y_i} pridedamas kampinis skirtumas m . Šis pakeitimas padeda pritraukti tai pačiai žymei priklausančių pavyzdžių vektorius arčiau žymės centro bei atitolinti juos nuo kitoms žymėms priklausančių vektorius, tiesiogiai paveikiant kampą tarp vektorius. Tai iliustruoja 5 paveikslėlis.



5 pav. Skirtumas tarp Softmax bei ArcFace paėmus 8 klases dvimatėje požymių erdvėje. Taškai rodo pavyzdžius, linijos – kiekvienos klasės centrą, prie kurio sutelkiami pavyzdžiai. Požymių vektorius normalizavimas išstumia juos ant apskritimo lanko. Nors Norm-Softmax išskaido taškus į grupes, matyti, kad skiriamosios sritys (angl. *decision boundaries*) ne tokios aiškios, kaip ArcFace nuostolių funkcijos atveju. [DGX⁺19].

2.6.1. SubcenterArcFace nuostolio funkcija

Naudojant ArcFace nuostolio funkciją daroma prielaida, kad mokymosi duomenys yra švarūs – ši nuostolio funkcija jautri triukšmingiems duomenims. Duomenys triukšmingi, kai paveikslėlių žymės sutampa, tačiau patys paveikslėliai tarpusavyje dažnai stipriai skiriasi nuo kitų (kitoks požiūrio kampas, papildomi objektai paveikslėlyje). Tai lemia klaidingai aukštas nuostolio funkcijos reikšmes ir trukdo mokymo procesui. Ši problema išsprendžiama SubcenterArcFace nuostolio funkcijoje pristatant galimybę požymių vektorius artinti ne prie vienintelio idealaus žymės pavyzdžio (klasės "centro") naudojant poklases (angl. *sub-classes*) – toliau nuo centro nutolusius pavyzdžius (6 pav.). Šia savybe galima pasinaudoti norint atmesti triukšmingus pavyzdžius, ir išvalyti mokymo duomenų rinkinį prieš naudojant ArcFace nuostolių funkciją neuroniniam tinklui apmokyti [DGL⁺20].



6 pav. Poklasių iliustracija [DGL⁺20].

SubcenterArcFace funkcija gaunama pakoregavus 12 lygtį:

$$\mathcal{L}_{SubcenterArcFace} = -\log \left(\frac{e^{s \cos(\theta_{y_i} + m)}}{e^{s \cos(\theta_{y_i} + m)} + \sum_{j=1, j \neq y_i}^N e^{s \cos(\theta_j)}} \right) \quad (13)$$

Žymėjimai:

- K - poklasių skaičius.
- θ_{y_i} - kampas tarp požymių vektoriaus x_i ir tikrosios klasės svorių vektoriaus W_{y_i} .
- θ_j - kampas tarp požymių vektoriaus x_i ir kitų klasių (ne tikrosios klasės) svorių vektorių W_{j_k} . Čia W_{j_k} j -toji svorių $W \in \mathbb{R}^{d \times N \times K}$, kur K - poklasių skaičius eilutė.
- s - mastelio koeficientas.

Paaiškinimas: Esminis skirtumas yra kampo tarp požymių vektoriaus ir klasių svorių vektorių parinkimas. Naudojama ši lygtis: $\theta_j = \arccos(\max_k W_{j_k}^T x_i)$, $k \in \{1, \dots, K\}$. Šis parinkimas užtikrina, kad per daug sudėtingi ir tolimiausiai nuo centro nutolusių pavyzdžių požymių vektoriai bus kiek atitolinami nuo arčiau teisingos klasės svorių vektoriaus esančių pavyzdžių.

2.7. Nuostolio funkcijų teorinis palyginimas

Šiame literatūros apžvalgos poskyryje pateikiamas trumpas pristatytų nuostolio funkcijų metodų palyginimas.



7 pav. Trejetų, Kontrasto, ArcFace bei SubcenterArcFace nuostolio funkcijų idėjų palyginimas.

7 pav. iliustruoja skirtumus tarp dalies nagrinėtų nuostolio funkcijų. Trejetų, Kontrasto ir SupCon nuostolio funkcijose atliekamas vietinis panašumo palyginimas: tarpusavyje lyginami pavyzdžiai priklausantys tam pačiam einamajam duomenų rinkiniui. Tuo tarpu ArcFace bei SubcenterArcFace atvejais atliekamas globalus palyginimas tarp pavyzdžio požymių vektoriaus ir klasės (ar poklasės) „centro“ – tos klasės svorių vektoriaus.

Skiriasi ir pristatytų nuostolio funkcijų skiriamosios sritys. Vietiniais palyginimais grįstos funkcijos (tolimesniuose skyriuose naudojamas „lokalios funkcijos“ trumpinys) neturi konkrečių formulėmis apibrėžtų skiriamųjų sričių. Trejetų, Kontrasto ir SupCon nuostolio funkcijų atveju skiriamąją sritimi tarp klasių tampa mokymo metu išmokstami Euklido atstumai tarp skirtingoms žymėms priklausančių paveikslukų požymių vektorių. Globaliais palyginimais grįstų funkcijų (čia ir toliau „globalių funkcijų“) atveju skiriamąsias sritis galima formaliai apibrėžti formulėmis. Iliustruoti skirtumus tarp skiriamųjų sričių 1 lentelėje pateikiamas funkcijų skiriamųjų sričių palyginimas dviejų žymių atveju.

1 lentelė. Globalių funkcijų skiriamosios sritys dviejų klasių atveju.

Nuostolių funkcija	Skiriamoji sritis
Softmax	$(\mathbf{W}_1 - \mathbf{W}_2) \cdot \mathbf{x} + b_1 - b_2 = 0$
Norm-Softmax	$\cos(\theta_1) = \cos(\theta_2)$
A-Softmax	K1: $\cos(m \cdot \theta_1) \geq \cos(\theta_2)$ K2: $\cos(m \cdot \theta_2) \geq \cos(\theta_1)$
CosFace	K1: $\cos(\theta_1) \geq \cos(\theta_2) + m$ K2: $\cos(\theta_2) \geq \cos(\theta_1) + m$
ArcFace	K1: $\cos(\theta_1 + m) \geq \cos(\theta_2)$ K2: $\cos(\theta_2 + m) \geq \cos(\theta_1)$

1 lentelėje pateiktose formulėse naudojami tie patys žymėjimai, kaip ir pateikti prie atitinkamų nuostolio funkcijų apžvalgos.

2.8. Trejetų bei porų išgavimo strategijos

Naudojant anksčiau apžvelgtas Trejetų Kontrasto bei SupCon nuostolio funkcijas yra svarbu, kaip mokymui parenkami trejetai ar poros. Didėjant pavyzdžių skaičiui, galimų mokymo porų skaičius auga polinomiškai [WHH⁺19]. Taip pat, ne visos poros yra tinkamos mokymo procesui – per lengvi pavyzdžiai, kai neigiamas pavyzdys jau yra toli nuo teigiamo pavyzdžio ir nuo atraminio vaizdo, nesuteikia daug informacijos ir nepadeda apmokyti neuroninio tinklo. Tam tikslui yra pasitelkiami sunkių teigiamų bei neigiamų porų išgavimo (angl. *mining*, *hard-negative/positive mining*) metodai.

Duomenų porų tipai:

- Lengvi neigiami (angl. *easy negatives*) pavyzdžiai – neigiamas pavyzdys yra toli nuo atraminio vaizdo, nuostolių funkcijos reikšmė lygi 0, nes $d(a,p) + \alpha < d(a,n)$.
- Pusiau sunkūs neigiami (angl. *semi-hard negatives*) pavyzdžiai – neigiamas pavyzdys yra toliau nuo atraminio vaizdo nei teigiamas, tačiau nuostolių funkcijos reikšmė vis dar teigiama. $d(a,p) < d(a,n) < d(a,p) + \alpha$
- Sudėtingi neigiami pavyzdžiai (angl. *hard negatives*) – neigiamas pavyzdys yra arčiau atraminio vaizdo, nei teigiamas $d(a,n) < d(a,p)$

Pavyzdžių išgavimas mokymo metu (angl. *online mining*) – tai metodas, kai sudėtingi teigiami ar neigiami pavyzdžiai parenkami tik iš einamosios iteracijos duomenų rinkinio.

Pavyzdžių išgavimas nemokymo metu (angl. *offline mining*) – šiuo atveju trejetai ar poros sudaromi apdorojant visą turimą duomenų rinkinį vienu metu ir parengiami dar prieš pradėdant mokymo procesą.

3. Eksperimentas

Norint atlikti apžvelgtų nuostolio funkcijų palyginimą nuspręsta apmokyti KNT paremtus modelius naudojant šias nuostolio funkcijas bei įvertinti jų pasiekiamus rezultatus.

3.1. Tyrimo eiga

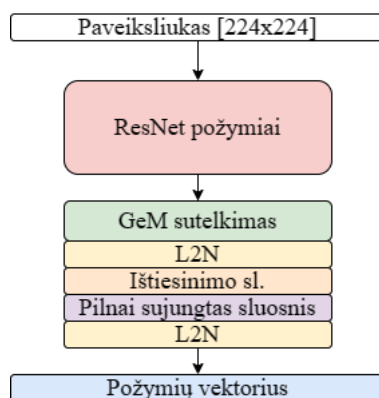
Prieš pradėdant eksperimentus atlikti tokie paruošiamieji darbai:

- Paruošti mokymo, validavimo bei testavimo duomenų rinkiniai – tai atlikta filtruojant „Google Landmarks Dataset v2“ mokymo duomenų poaibį, paliekant tik klases turinčias nuo 30 iki 50 pavyzdžių.
- Pasirinkti modelių gylyai – nuspręsta naudoti ResNet-18, ResNet-50 bei ResNet-101 tinklus. Mažiausias tinklas pasirinktas dėl greitesnio veikimo ir trumpiau trunkančio mokymo proceso, gilesni tinklai pasirinkti remiantis jų populiarumu.
- Pasirinktos duomenų augmentacijos.
- Pasirinktas optimizatorius „Adam“.
- Nuspręsta dėl galutinės modelio architektūros. Galutinė modelio architektūra pasirinkta remiantis [RTC18].
- Pasirinktos metrikos modeliams testuoti.
- Paruošti duomenys modelių testavimui – atlikta parsisiunčiant oxford5k bei paris6k duomenų rinkinius bei reikiamus žymių failus (angl. *ground truth*).

Eilinio eksperimento paleidimo eiga:

1. Pasirenkama nuostolio funkcija;
2. Lokalių palyginimų funkcijų atveju pasirenkama trejetų/porų parinkimo strategija (trejetai ar poros visada parenkami mokymo metu);
3. Pasirenkamas atitinkamo gylio tinklas (ResNet-18, ResNet-50 arba ResNet-101), naudojami jau apmokyti svoriai, inicializuojamas modelis;
4. Nurodomas modelio pavadinimas, epochų skaičius;
5. Jeigu reikia mokymas pratęsiamas nuo tarpinio taško;
6. Modelio mokymas paleidžiamas naudojant paruoštą skripto failą.

3.1.1. Naudota modelių architektūra



8 pav. Eksperimentuose naudotų modelių „GeMResNet“ struktūros schema.

8 pav. pavaizduota modelio architektūra pasirinkta remiantis [RTC18] straipsniu. Konvoliuciniai modelio sluoksniai, skirti požymių vektorių ištraukimui inicializuojami naudojant apmokytus skirtingo gylio ResNet modelių sluoksnius - iš esmės ResNet neuroninis tinklas panaudojamas kaip fundamentalus modelis požymiams ištraukti (angl. *backbone*). Esminiai pakeitimai atlikti paskutiniuose sluoksniuose. Vietoj sutelkimo vidurkinant sluoksnio (angl. *Average pooling*) naudojamas GeM sutelkimo sluoksnis. Pradinė GeM sluoksnio parametro p reikšmė 3, parametras bendras visoms paskutinio konvoliucinio sluoksnio išvesties dimensijoms. Jį seka L2 normalizavimo, vienas pilnai sujungtas ir vėl L2 normalizavimo sluoksniai. Modelių išvesties dimensija nevienoda - ResNet-18 naudojančių modelių išvesties dimensija 512, gilesnių modelių (ResNet-50 bei ResNet-101) išvesties dimensija 2048.

3.1.2. Naudotos priemonės

Modelių mokymo bei testavimo kodas parašytas JupyterHub aplinkoje Python programavimo kalba, modelių architektūrai bei mokymui aprašyti naudotas PyTorch karkasas.

Pasirinkta modelio architektūra paremta [RTC18] straipsnio autorių pateikta modelio struktūra bei GeM ir L2 normalizavimo sluoksnių implementacijomis⁶.

Modelių mokymo procese naudotų nuostolių funkcijų implementacijos imtos iš `pytorch-metric-learning`⁷ bibliotekos. Naudotasi ir šioje bibliotekoje pateiktomis duomenų išgavimo funkcijomis. Modelių testavimui reikalingų duomenų rinkinių atsisiuotimo bei modelių testavimo kodas parašytas remiantis kodu⁸ parengtu [RIT⁺18] straipsnio autorių. HOW metodo rezultatai gauti pasinaudojus [DTP⁺21] straipsnio autorių pateiktomis komandomis bei instrukcijomis⁹.

Modelių mokymui bei testavimui reikalingi skaičiavimai atlikti pasinaudojus suteiktais HPC ištekliais. Skaičiavimai atlikti ant NVIDIA Tesla V100 32GB grafines plokštes naudojančio GPU klasterio. Visiems eksperimentams atlikti prireikė 291 GPU skaičiavimo valandos.

⁶<https://github.com/filipradenovic/cnnimageretrieval-pytorch/blob/cirtorch/layers/pooling.py#L36>

⁷<https://github.com/KevinMusgrave/pytorch-metric-learning>

⁸<https://github.com/filipradenovic/revisitop>

⁹https://github.com/facebookresearch/isc2021/blob/main/docs/6_how_subset.md

3.2. Duomenys

3.2.1. „Google Landmarks Dataset v2“

„Google Landmarks Dataset v2“ (GLDv2) yra šiuo metu didžiausias įvairiausių žymių vietų (angl. *landmarks*) paveikslėlių rinkinys, skirtas žymių vietų atpažinimo bei atrinkimo uždaviniams spręsti [WAC⁺20]. Pagrindinis paveikslėlių šaltinis yra Wikimedia Commons¹⁰ – Vikipedijos medijos duomenų saugykla. Joje esantys duomenys turi atvirojo turinio (angl. *Creative Commons*), bei viešosios srities (angl. *Public Domain*) licencijas.

GLDv2 pasižymi keliomis iššūkių neuroniniams tinklams keliančiomis savybėmis:

- Didžiulis dydis: virš penkių milijonų paveikslėlių, priklausančių daugiau nei dviems šimtams klasių.
- Klasių disbalansas: 57% klasių turi iki dešimties pavyzdžių, o 38% klasių turi iki penkių paveikslėlių.
- Tikslui nepriklausantys vaizdai (angl. *out-of-domain images*): užklausų poaibyje jų net (98,9%). Siekiant sukurti kuo realistiškesnį duomenų rinkinį, pridėta ir ne žymių vietų vaizduojančių paveikslėlių. Taip skatinant kurti patikimesnius (teikiančius nedidelį klaidingai teigiamų rezultatų skaičių) modelius.
- Variacija klasių viduje (angl. *intra-class variation*): tos pačios klasės paveikslėliai gali būti labai skirtingi – juose vaizduojama žymi vieta iš skirtingų pusių, perteikiant skirtingas detales, iš vidaus ar išorės (pastato atveju).

GLDv2 yra labai triukšmingas duomenų rinkinys, tačiau galima rasti viešai prieinamą ne tokį triukšmingą GLDv2_train_clean duomenų rinkinio poaibį. Poaibis paruoštas automatiškai būdu išvalant pagrindinį duomenų rinkinį – atrenkant paveikslėlius, kurie per daug skiriasi nuo kitų klasės pavyzdžių. Apie poaibio paruošimą daugiau detalių pateikiama [YOS⁺20] straipsnyje.

3.2.2. „Oxford Buildings“ ir „Paris Dataset“

„Oxford Buildings“ (toliau trumpinama oxford5k) yra duomenų rinkinys, sudarytas iš penkių tūkstančių paveikslėlių, vaizduojančių įvairius Oxfordo universiteto pastatus. Atitinkamai „Paris Dataset“ (toliau trumpinama paris6k) – į oxford5k labai panašus, šešis tūkstančius paveikslėlių, vaizduojančių įvairius Paryžiaus miesto pastatus, turintis duomenų rinkinys.

Šie rinkiniai skirti vaizdų atrinkimo (angl. *image retrieval*) uždavinį sprendžiantiems modeliams vertinti. Abu rinkiniai turi 55 užklausų paveikslėlius, po penkis kiekvienai iš 11 žymių vietų taip sudarant 11 užklausų grupių. Paveikslėliai pagal tinkamumą skirstomi į tris žymes kiekvienoje užklausų grupėje: teigiama, kai paveikslėliuke matyti daugiau nei 25 procentai objekto, neigiama jei paveikslėliuke nėra užklausos objekto ir nenaudinga (angl. *junk*), kai matyti mažiau nei 25 procentai objekto. Modelių gražinamų rezultatų tikslumas matuojamas skaičiuojant vidutinio preciziškumo vidurkį (mAP) per visas 55 užklausas, ignoruojant nenaudingus paveikslėlius. Tikslas yra grąžinti visus duomenų rinkinio paveikslėlius, rodančius tą patį objektą.

¹⁰<https://commons.wikimedia.org>

3.2.3. „Revisited Oxford“ ir „Revisited Paris“

Peržiūrėti oxford5k ir paris6k duomenų rinkiniai (čia ir toliau trumpinama roxford5k bei rparis6k) yra iš naujo sužymėti oxford5k bei paris6k duomenų rinkiniai, pristatyti [RIT⁺18] straipsnyje. Peržiūrėjimo metu ištaisytos originalių duomenų sužymėjimo klaidos, pristatyta 15 naujų sudėtingų užklausių.

Naujas sužymėjimas paveikslukus suskirsto 4 žymes:

- Lengvi (angl. *Easy*): paveiksluke aiškiai matyti ieškomas objektas.
- Sunkūs (angl. *Hard*): galima atpažinti objektą, tačiau jį sudėtinga sutapatinti su užklaustos paveiksluku.
- Neaiškūs (angl. *Unclear*): paveikslukas galimai vaizduoja užklaustos objektą, tačiau trūksta vizualinio konteksto. (Pavyzdžiui: paveiksluke vaizduojama kita dalinai simetriško objekto pusė, nei užklaustos paveiksluke todėl neaišku, ar jame yra ieškomas objektas).
- Neigiami (angl. *Negative*): paveikslukas neatitinka kitų kriterijų.

Ir pristato tris naujus vertinimo metodus:

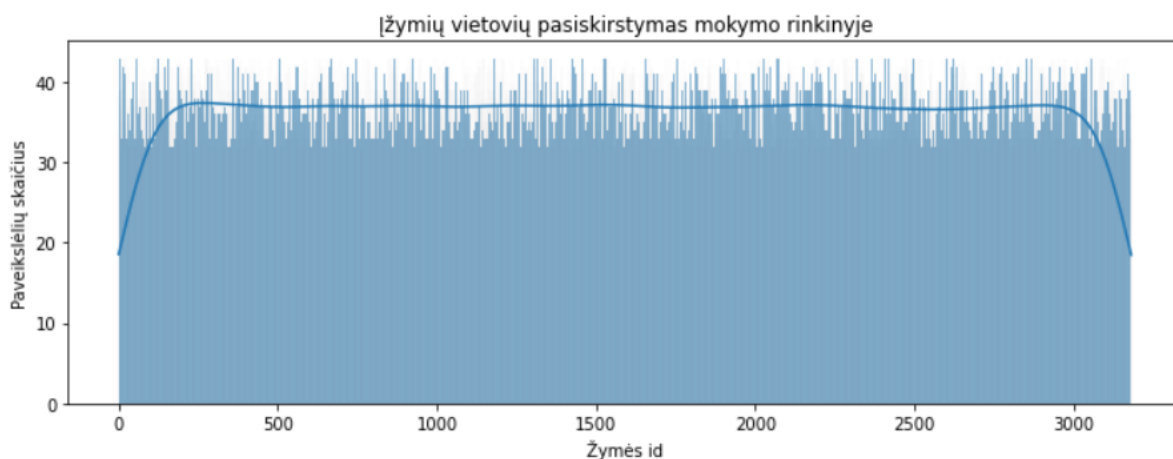
- Lengvas (E) (angl. *Easy*): lengvi paveikslukai laikomi teigiamais pavyzdžiais, sunkūs ir neaiškūs ignoruojami.
- Vidutinis (M) (angl. *Medium*): lengvi ir sunkūs laikomi teigiamais pavyzdžiais, neaiškūs ignoruojami.
- Sunkus (H) (angl. *Hard*): tik sunkūs paveikslukai laikomi teigiamais pavyzdžiais, lengvi ir neaiškūs ignoruojami.

Straipsnyje pristatomas ir vieną milijoną turintis nukreipimo paveikslukų duomenų rinkinys „R1M“ skirtas didelio masto eksperimentams. Jis puikiai tinka išsamiam modelių testavimui, tačiau šiame eksperimente jis nenaudotas, dėl mažesnio tyrimo masto.

3.3. Eksperimente naudoti duomenų rinkinių poaibiai

Modeliams apmokyti naudotas duomenų rinkinys (čia ir toliau gldv2_mokymo) gautas filtruojant išvalytą GLDv2 duomenų rinkinio versiją (GLDv2_train_clean). Filtruota pasirenkant tik tas įžymias vietas, kurios turi nuo 30 iki 50 paveikslėlių. Gautą mokymo duomenų poaibį sudaro beveik 118 tūkstančių paveikslėlių (117 847), priklausančių 3 182 žymėms. Mokymo duomenų pasiskirstymo grafikas pateiktas 9 paveikslėlyje. Duomenys filtruoti ir atsisiųsti naudojantis Kaggle platforma¹¹.

¹¹<https://www.kaggle.com/competitions/landmark-recognition-2020>



9 pav. Mokymo duomenų pasiskirstymo grafikas.

Filtruojant GLDv2_train_clean paruošti validavimo ir testavimo duomenų rinkiniai (čia ir toliau trumpinami gldv2_validavimo bei gldv2_testavimo). Validavimo duomenų rinkinys naudotas mokymo proceso stebėjimui, taip pat, kartu su testavimo duomenų rinkiniu, modelių vertinimo metu. Kartu sudėjus gldv2_validavimo ir gldv2_testavimo poaibius gaunamas gldv2_tiny rinkinys, naudojamas testavimo metu. Validavimo poaibio pavyzdžiai naudojami kaip užklaustos, o testavimo - kaip paieškos pavyzdžiai (angl. *reference set*).

2 lentelė. Eksperimente modelių mokymo ir testavimo metu naudoti duomenų rinkiniai.

Duomenų rinkinys	Paveikslėlių skaičius	Źymių vietų skaičius	Pavyzdžių skaičius Źymei
gldv2_mokymo	117847	3182	32-43
gldv2_validavimo	13784	3182	4-5
gldv2_testavimo	6203	3182	1-2
paris6k	6392	11	5
oxford5k	5063	11	5

2 lentelėje pateikiami visi eksperimento metu naudoti duomenų rinkiniai, kartu su pavyzdžių bei Źymių skaičiumi. Čia neminimi roxford5k ar rparis6k nes naudojami tie patys paveikslėliai, kaip ir oxford5k bei paris6k rinkiniuose.

3.3.1. Duomenų augmentacijos

Mokymo vaizdams pritaikytos augmentacijos, naudota duomenų augmentacijų biblioteka Albumentations¹². Buvo pasirinktos ir pritaikytos šios duomenų augmentacijos:

1. Atsitiktinis iškirpimas [224x224],
2. Atsitiktinis horizontalus apvertimas,
3. Atsitiktinis ryškumo, kontrasto, atspalvio ir paveikslėlio saturacijos reikšmių pakeitimas,
4. Atsitiktinis pasukimas 10 laipsnių.

¹²<https://github.com/albumentations-team/albumentations>

Testavimo vaizdams apdoroti naudotos tik ResNet tipo neuroniniams tinklams reikiamos transformacijos¹³:

1. Dydžio pakeitimas į [224x224],
2. Pikselių reikšmių normalizavimas,
3. Pavertimas į tenzorių.

3.4. Naudotos trejetų ir porų išgavimo strategijos

Modelių mokymo metu naudotos dvi trejetų bei porų išgavimo mokymo metu strategijos.

3.4.1. Trejetų išgavimo strategija

Trejetų, reikalingų Trejetų nuostolio funkcijai, išgavimui buvo naudota pytorch-metric-learning bibliotekos funkcija „BatchHardMiner“¹⁴ (trumpinama BHM), realizuota vadovaujantis [XSP20]. Naudotas sudėtingų teigiamų ir sudėtingų neigiamų pavyzdžių išgavimas. Atrenkami pavyzdžiai apibrėžiami taip:

Sunkūs neigiami pavyzdžiai - tai į atraminį vaizdą panašiausi pavyzdžiai, kurių žymė nesutampa su atraminio vaizdo žyme. (14) formulėje x_{sn} - sunkus neigiamas pavyzdys, x_a - atraminis vaizdas, x - visi einamajame rinkinyje esantys vaizdai, kurių žymė nesutampa su atraminio vaizdo žyme.

$$x_{sn} = \operatorname{argmin}(d(f(x_a), f(x))) \quad (14)$$

Sunkūs teigiami pavyzdžiai - tai į atraminį vaizdą mažiausiai panašūs pavyzdžiai, turintys tą pačią žymę kaip ir atraminis vaizdas. (15) formulėje x_{st} - sunkus teigiamas pavyzdys, x - visi einamajame rinkinyje esantys vaizdai, kurių žymė sutampa su atraminio vaizdo žyme.

$$x_{st} = \operatorname{argmax}(d(f(x_a), f(x))) \quad (15)$$

3.4.2. Porų išgavimo strategija

Porų išgavimui naudotasi tos pačios pytorch-metric-learning bibliotekoje esančia „MultiSimilarityMiner“¹⁵ (trumpinama Msim) funkcija. Jos implementacija paremta [WHH⁺19]. Straipsnyje pateikiami trys skirtingi panašumo vertinimo būdai:

- Poros panašumas (angl. *self-similarity*): panašumas tarp paveikslėlių poroje, skaičiuojamas naudojant kosinuso panašumo formulę (3).
- Neigiamas santykinis panašumas (angl. *negative relative similarity*): panašumas tarp neigiamo pavyzdžio ir gretimų neigiamų pavyzdžių.

¹³Pagal Torchvision dokumentaciją: <https://pytorch.org/torchvision/models/resnet101.html>

¹⁴<https://kevinmusgrave.github.io/pytorch-metric-learning/miners/#batchhardminer>.

¹⁵<https://kevinmusgrave.github.io/pytorch-metric-learning/miners/#multisimilarityminer>.

- Teigiamas santykinis panašumas (angl. *positive relative similarity*): panašumas tarp atraminio vaizdo ir gretimų teigiamų pavyzdžių. Kuo teigiami pavyzdžiai arčiau vienas kito, tuo mažiau informacijos tokios poros perneša.

Msim funkcijoje naudojamas neigiamo–teigiamo santykinio panašumo palyginimas:

- Neigiamos poros parenkamos, kai jų panašumas didesnis už sunkiausios (su mažiausiu atraminio ir teigiamo vaizdo tarpusavio panašumu) teigiamos poros panašumą, iš jo atėmus ϵ (nedidelę teigiamą reikšmę turintį skirtumo parametą).
- Teigiamos poros parenkamos jei jų panašumas mažesnis nei sudėtingiausios neigiamos poros panašumas, prie jo pridėjus ϵ .

Eksperimentams parinkta ϵ parametro reikšmė – 0,1.

3.5. Modelių mokymo procesas

Neuroninio tinklo mokymo procesas vyksta taip: mokymo duomenys suskirstomi į einauosius duomenų rinkinius (pasirinktas rinkinio dydis 64). Vienas visų duomenų rinkinių apdorojimas vadinamas epocha. Einamojo rinkinio paveikslukai kartu su žymėmis paduodami nuostolių funkcijai, paskaičiuojamos nuostolių funkcijų reikšmės, jomis remiantis atnaujinami svoriai. Svorius atnaujina optimizatorius, didžiulę įtaką mokymo procesui daro mokymo žingsnio parametras, kuriuo kontroliuojamas mokymo greitis – nuo jo didumo priklauso, kiek einamojo rinkinio metu paskaičiuotos nuostolio reikšmės keis svorių reikšmes. Mokymas kartojamas numatytą epochų skaičių.

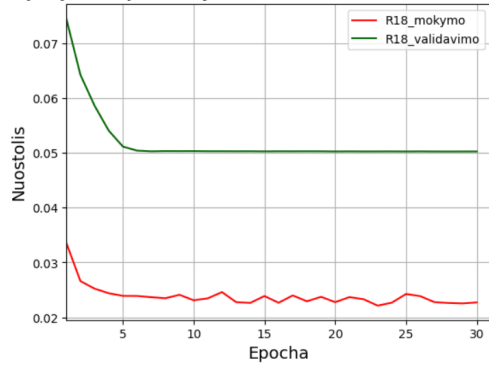
3.5.1. Hiperparametrų parinkimas

Modelių mokymo procesas ir pasiekiami rezultatai yra jautrūs mokymo hiperparametrms – epochų skaičiui bei mokymo žingsnio reikšmei. Norint išsiaiškinti reikiamas hiperparametrų reikšmes reikia atlikti bandymus ir stebėti mokymo procesą. Priimti sprendimus galima stebint nuostolio funkcijos reikšmių kitimo mokymo proceso metu grafikai. Stebimos nuostolio reikšmės imant mokymo ir validavimo duomenų poaibius. Tai padeda ne tik nustatyti hiperparametrų reikšmes, bet ir parodo kada modelis persimoko arba konverguoja (pasiekia minimumą).

3.5.1.1. Epochų skaičiaus parinkimas

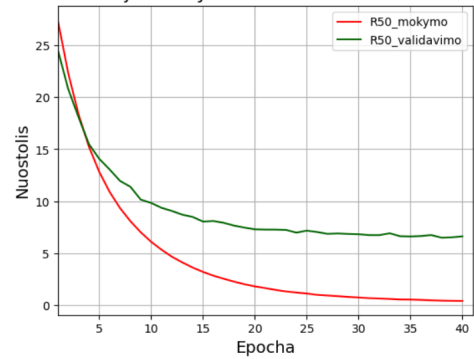
Gauti mokymo proceso grafikai padėjo parinkti tinkamas hiperparametrų reikšmes. Pavyzdžiui, pirma visi modeliai buvo mokyti 30 epochų, tačiau pastebėjus, kad nepasiekiamas mažiausia galima nuostolio reikšmė (žr. 10b pav.) buvo pasirinktas didesnis epochų skaičius (40 epochų). Kaip matyti iš 10 pav. su Trejetų nuostolio funkcija buvo kiek kitaip – pastebėta, kad modelis konverguoja po 15-kos epochų, todėl mokant gilesnius modelius buvo pasirinktas mažesnis epochų skaičius. Kai kuriais atvejais mokymo grafikai leido daryti išvadą ar mokymo procesas buvo sėkmingas ar ne, pavyzdžiui, iš 10c pav. matyti, kad mokymasis buvo sėkmingas (nuosekliai mažėjo nuostolio reikšmės).

Trejetų funkcijos mokymo ir validavimo nuostolio reikšmės



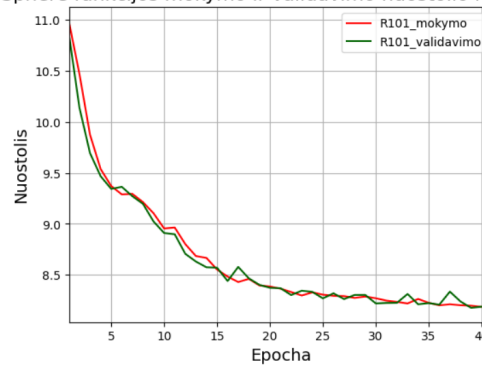
(a) R18-Trejetų

CosFace funkcijos mokymo ir validavimo nuostolio reikšmės



(b) R50-CosFace

Sphere funkcijos mokymo ir validavimo nuostolio reikšmės



(c) R101-SphereFace

10 pav. Modelių mokymo proceso grafikai naudojant Trejetų, CosFace bei SphereFace nuostolio funkcijas.

3.5.1.2. Mokymo žingsnio bei jo planuotojo parinkimas

KNT neuroninių tinklų mokyme naudojant „Adam“ optimizatorių dažniausiai naudojama pradinė mokymo žingsnio reikšmė yra 10^{-3} . Remiantis [RTC18] straipsniu bandytos kiek mažesnės mokymo žingsnio reikšmės – 10^{-4} , 10^{-6} , tačiau mokymo proceso grafikai parodė, kad jos per mažos efektyviam mokymui, todėl galutinė mokymo žingsnio reikšmė liko 10^{-3} .

Eksperimentuojant buvo keistas ir mokymo žingsnio planuotojas (angl. *learning rate scheduler*) – pirma bandyta naudoti „ReduceLRonPlateau“¹⁶ planuotoją. Juo naudojantis mokymo žingsnis mažinamas, tik nuostoliui nekintant numatytą epochų skaičių. Ši strategija neturėjo didelės įtakos mokymui, dėl palyginus nedidelio epochų skaičiaus, todėl vėliau pasirinkta „ExponentialLR“¹⁷ mokymo žingsnio kitimo strategija, kiekvieną epochą sumažinanti mokymo žingsnį padauginant jį iš $e^{-0,01}$.

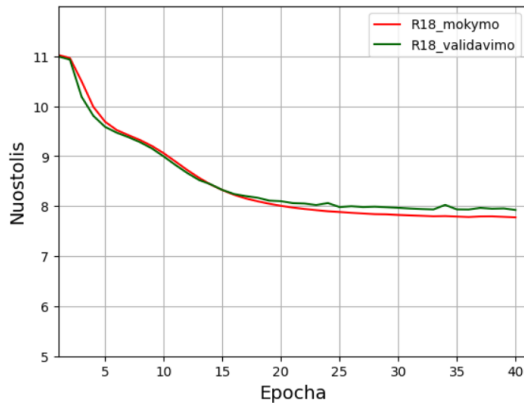
3.5.1.3. Duomenų išgavimo naudojimas

Remiantis nuostolio funkcijų idėjomis trejetų ar porų išgavimo reikalauja tik lokali nuostolio funkcijos. Naudojant jas aktualu mokymo metu pateikti sudėtingus ir informatyvius pavyzdžius, atrinktus įvertinant Euklido atstumą tarp einamojo duomenų rinkinio pavyzdžių. Kadangi globalių

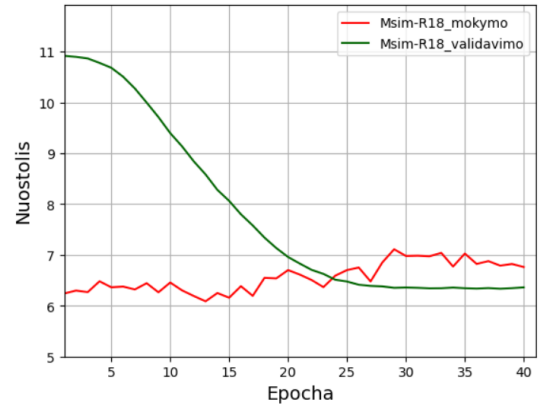
¹⁶https://pytorch.org/docs/stable/generated/torch.optim.lr_scheduler.ReduceLRonPlateau.html.

¹⁷https://pytorch.org/docs/stable/generated/torch.optim.lr_scheduler.ExponentialLR.html.

funkcijų skiriamosios sritys didesnę svarbą skiria kampui tarp požymių vektorių, atlikti informatyvių porų atrinkimą nėra poreikio. Vis dėlto, norint sužinoti, kaip toks parinkimas paveikia rezultatus, pabandyta ši metodą įtraukti į globalias funkcijas naudojančių modelių mokymo procesą. Gauti mokymo proceso grafikai rodo, kaip duomenų išgavimas reikšmingai paveikia nuostolio kitimą, kartu ir gaunamus rezultatus.



(a) SphereFace be duomenų atrinkimo



(b) SphereFace su duomenų atrinkimu

11 pav. R18-GeM-SphereFace mokymo proceso grafikų skirtumas su ir be duomenų atrinkimo mokymo metu.

3.6. Vertinimo metrikos

Norint įvertinti apmokytų modelių gražinamų rezultatų tikslumą naudojamos įvairios vertinimo metrikos. Dalis jų susijusios su modelio tikslumo įverčiu (įvertina ar modelis tinkamai susieja paveiksluką su jo žyme). Kitos metrikos skirtos vaizdų atrinkimo modeliams vertinti: pateikus užklauso paveiksluką vertinamas modelio atrinktų ”panašių” paveikslukų gerumas – gražintų rezultatų eiliškumas bei žymių teisingumas. Eksperimento metu gautiems modeliams įvertinti buvo naudotos šios metrikos:

- Preciziškumas K range – Preciziškumas@K.
- Preciziškumo vidurkis K range (angl. *mean Precision at rank K* – $mP@K$)
- Vidutinio preciziškumo vidurkis (angl. *mean Average Precision*), trumpinamas mAP. Mokslinėje literatūroje dar vadinamas makro preciziškumo vidurkiu (angl. *macro Average Precision*).
- Vidutinis atvirkštinis rangas (angl. *Mean Reciprocal Rank* – MRR).

Preciziškumas@K (angl. *Precision@K*) – tai užklauso gražintų rezultatų tikslumo įvertis, paskaičiuotas atsižvelgiant tik į pirmus K rezultatus. Jis parodo, kiek vienos užklauso rezultatų iš pirmų K rezultatų yra teisingi – ar jų žymė sutampa su užklauso paveiksluko žyme.

$$\text{Preciziškumas@K} = \frac{\text{Teisingi rezultatai iš pirmų K rezultatų}}{K} \quad (16)$$

Preciziškumo vidurkis K range (angl. *mean Precision at rank K - mP@K*) – tai preciziškumo@K vidurkis, paskaičiuotas susumavus preciziškumo pirmose K pozicijose rezultatus per visas užklausas ir padalinus iš užklausų skaičiaus. Šis įvertis parodo, kiek tikslūs vidutiniškai yra pirmi K užklausos rezultatai.

$$mP@K = \frac{1}{N} \sum_{i=1}^N Tikslumas@K_i \quad (17)$$

Vidutinio preciziškumo vidurkis (mAP) – metrika, plačiai naudojama vertinant informacijos paieškos sistemas, objektų aptikimo modelius. Ji parodo, kaip gerai modelis įvykdo įprastą užklausą atsižvelgiant į grąžintų rezultatų aktualumą (šiuo atveju panašumą į užklausos paveikslėlį) bei aktualių rezultatų poziciją.

$$mAP = \frac{1}{Q} \sum_{q=1}^Q AP(q) \quad (18)$$

Vidutinis preciziškumas@K (AP@K) – tai vidurkis Preciziškumas@K reikšmių paskaičiuotų pirmose K pozicijose, į vertinimą įtraukiant tik tas pozicijas, kuriose rezultatas yra teisingas. (19) formulė naudojama vidutinio preciziškumo vidurkiui apskaičiuoti.

$$AP = \frac{1}{N} \sum_{k=1}^K Preciziškumas@k \cdot rel(k) \quad (19)$$

Žymėjimai:

- N - teisingų rezultatų skaičius.
- K - dominančių rezultatų skaičius.
- $rel(i)$ - funkcija, lygi 1 jei rezultatas pozicijoje i yra teisingas ir 0 kitu atveju.

Vidutinis atvirkštinis rangas (MRR) – tai metrika vertinanti grąžinamų aktualių rezultatų eiliškumu. Ji atsako į klausimą kurioje pozicijoje vidutiniškai yra teisingiausias rezultatas tarp K gautų rezultatų. Metrikos reikšmė lygi 1, kai pirmas grąžintas variantas visada yra tas, kurio ieškoma. Šiuo atveju parodo, kurioje pozicijoje vidutiniškai bus pirmas užklausos paveikslėlio žymę atitinkantis paveikslėlis.

$$MRR = \frac{1}{N} \sum_{i=1}^N \frac{1}{\text{pozicija}_i} \quad (20)$$

Žymėjimai:

- N - užklausos rezultatų skaičius.
- pozicija_i - pirmo tikrai teisingo pavyzdžio pozicija tarp gautų N rezultatų.

3.7. Eksperimento rezultatai

3.7.1. Nuostolių funkcijų pritaikymo palyginimas – mAP įvertis

Rezultatų lentelėse naudojamų žymėjimų paaiškinimas:

- Modelio pavadinimas: prasideda trumpiniu, nurodančiu kokio gylio modelis naudotas. Jei gū nėra naudojamas duomenų išgavimas, toliau seka mokymo metu naudotos nuostolių funkcijos pavadinimas, pavyzdžiui: "R101-GeM-Trejetų". Kitu atveju pradžioje nurodoma duomenų išgavimo funkcija (Msim arba BHM).
- Spalvos:
 - Mėlyna: išskiria aukščiausius rezultatus pasiekusį modelį, mokytą naudojant lokalią nuostolių funkciją.
 - Žalia: išskiria geriausią modelį mokytą su globalia nuostolių funkcija.
 - Raudona: papildomai nemokyto ResNet modelio be paskutinio sluoksnio rezultatai.
 - Geltona: HOW metodu paremtas modelis.
- Paryškinti rezultatai: bendrai geriausias modelis to paties gylio modelių grupėje.

3 lentelė. Galutiniai mAP rezultatai oxford5k, paris6k duomenims. Pateikiami ir trijų vertinimo protokolų – lengvo (E), vidutinio (M) bei sunkaus (H) įverčiai roxford5k bei rparis6k duomenų rinkiniams.

Modelis	oxford5k	paris6k	roxford5k mAP %			rparis6k mAP %		
	mAP %	mAP %	E	M	H	E	M	H
R101-GeM-Trejetų	10,96	24,24	6,18	5,93	1,19	22,83	19,37	7,42
R101-GeM-SupCon	20,04	33,22	11,56	9,83	2,16	29,30	26,01	11,18
R101-GeM-ArcFace	16,89	33,91	14,19	12,99	7,13	32,41	25,77	10,61
R101-GeM-SubCArcFace	8,19	21,15	4,17	4,38	1,10	17,88	18,44	9,06
R101-GeM-CosFace	25,43	44,86	19,87	16,04	6,93	42,04	34,00	15,64
R101-GeM-SphereFace	4,39	6,52	3,04	3,50	1,02	4,91	7,29	3,73
R101-Identity	6,15	25,79	4,62	5,70	2,12	23,80	24,68	13,20
R50-GeM-Trejetų	13,57	26,88	6,72	6,26	1,22	24,91	21,08	7,97
R50-GeM-SupCon	21,59	36,84	15,02	11,24	2,22	34,92	28,96	12,93
R50-GeM-ArcFace	23,08	40,90	17,86	12,86	2,29	35,84	29,88	12,48
R50-GeM-SubCArcface	23,98	41,34	17,00	13,78	4,16	37,89	30,35	11,62
R50-GeM-CosFace	27,61	45,01	22,76	17,95	6,68	43,42	32,84	13,88
R50-GeM-SphereFace	2,51	7,43	2,16	2,74	1,03	5,99	7,25	3,11
R50-Identity	4,83	17,28	3,62	4,88	2,12	16,12	17,15	7,98
R18-GeM-Trejetų	3,01	5,94	2,48	2,88	0,91	4,56	6,71	3,21
R18-GeM-SupCon	13,88	23,10	7,97	7,62	1,94	19,98	20,12	9,43
R18-GeM-ArcFace	19,60	41,53	12,25	9,13	1,12	38,60	32,02	14,57
R18-GeM-SubCArcFace	26,17	40,53	15,65	11,48	1,73	36,87	29,51	11,64
R18-GeM-CosFace	22,18	40,26	15,62	11,75	2,13	37,59	29,98	11,63
R18-GeM-SphereFace	4,19	6,45	3,16	3,65	1,17	5,06	6,82	3,26
R18-Identity	3,78	13,54	3,21	4,01	1,36	12,72	13,66	6,47
HOW-R18-1000	-	-	92,59	75,24	51,05	93,02	79,76	58,84

Pastebėjimai ir 3 lentelės rezultatų apibendrinimai:

- Bendrai aukščiausius rezultatus pasiekė ResNet-50 architektūra paremti modeliai (R50-GeM).
- Tarp lokalių funkcijų (Trejetų bei SupCon) geresnius rezultatus pasiekia SupCon nuostolių funkcija.
- Tarp globalių funkcijų (ArcFace, SubcenterArcFace, CosFace, SphereFace) geriausių rezultatų pateikia CosFace bei SubcenterArcFace nuostolių funkcijos.
- SphereFace nuostolių funkcijos rezultatai prasčiausi neatsižvelgiant į pasirinkto modelio gylį ar funkcijos tipą.
- Nė vienas modelis nepasiekia specializuoto HOW metodo pasiekiamų rezultatų.

Pastaba: mAP, MRR bei P@1 metrikų įverčius gldv2_tiny duomenų rinkiniui galima rasti priede nr. 2.

3.7.2. Nuostolių funkcijų pritaikymo palyginimas - mP@K įvertis

Žemiau pateiktoje rezultatų lentelėje naudojami tie patys žymėjimai, kaip ir aukščiau pateiktoje 3 lentelėje.

4 lentelė. Galutiniai mP@K rezultatai kai $K = [1, 5, 10]$ su roxford5k duomenų rinkiniu. Rezultatai pateikiami trims anksčiau minėtiems vertinimo protokolams - E, M, H.

Modelis	roxford5k E %			roxford5k M %			roxford5k H %		
	mP@1	mP@5	mP@10	mP@1	mP@5	mP@10	mP@1	mP@5	mP@10
R101-GeM-Trejetų	13,24	13,82	12,50	12,86	14,57	13,29	1,43	2,86	2,29
R101-GeM-SupCon	16,18	16,25	15,66	17,14	15,71	15,05	7,14	4,57	4,00
R101-GeM-ArcFace	20,59	20,00	19,41	20,00	20,57	19,86	7,14	7,71	7,71
R101-GeM-SubCArcFace	8,82	7,65	7,21	8,57	8,29	7,86	< 0,00	1,14	1,29
R101-GeM-CosFace	29,41	25,29	23,49	30,00	26,38	23,24	10,00	10,14	8,86
R101-GeM-SphereFace	8,82	7,35	5,88	8,57	8,00	6,57	1,43	1,43	1,14
R101-Identity	4,41	3,24	3,38	4,29	4,86	4,86	1,43	2,57	1,86
R50-GeM-Trejetų	14,71	12,35	11,77	15,71	12,57	12,14	2,86	1,14	1,86
R50-GeM-SupCon	27,94	21,52	18,87	30,00	20,00	16,57	10,00	5,71	6,14
R50-GeM-ArcFace	25,00	26,62	25,48	25,71	22,29	20,91	7,14	4,29	3,71
R50-GeM-SubCArcface	25,00	23,09	22,61	27,14	21,14	20,41	8,57	5,43	4,78
R50-GeM-CosFace	30,88	27,65	26,30	31,43	27,38	25,81	7,14	8,91	8,91
R50-GeM-SphereFace	2,94	2,94	2,50	4,29	4,00	3,43	2,86	1,14	1,14
R50-Identity	< 0,00	1,18	1,62	0,00	7,43	7,86	< 0,00	6,86	6,86
R18-GeM-Trejetų	4,41	4,41	4,12	4,29	5,14	4,57	< 0,00	1,14	0,86
R18-GeM-SupCon	19,12	14,90	13,68	22,86	18,57	16,00	7,14	6,57	4,43
R18-GeM-ArcFace	19,12	16,77	15,33	18,57	17,43	15,43	1,43	2,29	2,29
R18-GeM-SubCArcFace	26,47	21,62	20,59	25,71	20,29	18,88	8,57	4,86	2,86
R18-GeM-CosFace	20,59	20,15	18,74	25,71	18,57	16,86	8,57	4,29	3,86
R18-GeM-SphereFace	4,41	4,41	3,97	5,71	6,00	5,14	1,43	2,29	1,71
R18-Identity	< 0,00	< 0,00	0,59	< 0,00	1,43	1,71	< 0,00	1,43	1,43

Pastebėjimai ir 4 lentelės rezultatų apibendrinimai:

- Bendri pastebėjimai sutampa su 3 lentelės iliustruojamais rezultatais - geriausia architektūra R50-GeM, prasčiausi rezultatai naudojant SphereFace nuostolių funkciją.
- Bendru atveju didžiulio mP@K skirtumo tarp E bei M vertinimo protokolų nėra, tačiau H vertinimo metodo rezultatai akivaizdžiai prastesni.
- Bendru atveju mP@1 rezultatai geresni nei mP@5 ar mP@10 - modeliai dažniau gerai parenka pirmuosius, panašiausius paveikslukus, tačiau esant sudėtingesniems pavyzdžiams tolimesni rezultatai neatitinka užklaustos paveiksluko žymės.

Priede nr. 1 pateikiami mP@K rezultatai su rparis6k duomenų rinkiniui. Panagrinėjus matyti, kad su rparis6k duomenimis rezultatai visada pastebimai aukštesni. Tai sutampa su [RIT⁺18] straipsnyje pateiktais rezultatais.

3.7.3. Duomenų atrinkimo pritaikymo tyrimas

Vykdamt modelių mokymą pastebėta, kad duomenų atrinkimo pritaikymas ir mokymo su globaliomis funkcijomis metu reikšmingai paveikia modelių rezultatus.

5 lentelė. Modelių mokytų naudojant duomenų atrinkimą rezultatai. Pateikiama dalis rezultatų – oxford5k, paris6k bei gldv2_tiny rinkiniams. roxford5k bei rparis6k rezultatai pateikiami priede nr. 3

Modelis	oxford5k	paris6k	gldv2_tiny		
	mAP %	mAP %	mAP %	MRR %	P@1 %
Msim-R101-GeM-ArcFace	19,32	41,48	74,67	83,87	82,84
Msim-R101-GeM-SubCArcFace	24,57	44,61	72,35	82,52	81,44
Msim-R101-GeM-CosFace	20,34	33,20	74,78	84,56	83,29
Msim-R101-GeM-SphereFace	34,84	57,09	12,84	21,85	19,85
R101-Identity	6,15	25,79	15,72	26,57	23,88
Msim-R50-GeM-ArcFace	26,99	44,23	73,57	83,34	81,97
Msim-R50-GeM-SubCArcFace	27,20	51,12	71,61	81,87	80,51
Msim-R50-GeM-CosFace	28,13	52,74	73,30	83,88	82,48
Msim-R50-GeM-SphereFace	33,86	58,54	13,35	22,69	20,45
R50-Identity	4,83	17,28	15,24	25,52	22,89
Msim-R18-GeM-ArcFace	30,57	51,61	50,52	64,90	62,26
Msim-R18-GeM-SubCArcF	30,43	46,29	49,67	64,09	62,05
Msim-R18-GeM-CosFace	31,85	50,81	49,49	64,78	61,93
R18-GeM-SphereFace	32,40	62,41	12,37	21,21	19,18
R18-Identity	3,78	13,54	13,00	22,71	20,31

Pastebėjimai ir 5 lentelėje pateiktų rezultatų apibendrinimai:

- Gauti rezultatai rodo, kad porų atrinkimo naudojimas bendru atveju pagerina modelių pasiekiamus rezultatus 1-10 procentų (išskirtys - R101-GeM-CosFace rezultatai suprastėja, SphereFace atveju mAP įverčiai padidėja virš 20 procentų).
- Naudojant duomenų atrinkimą bendrai (tarp visų funkcijų) geriausias rezultatus tiek mAP tiek mP@K įverčių, per visus duomenų rinkinius (žr. roxford ir rparis mP@K rezultatus priede nr. 4) pasiekia ResNet-18 gylio tinklai.
- Patį geriausią rezultatą per visus gylius rodo su SphereFace mokytai modeliai. Tarp jų geriausias: R101-GeM-SphereFace modelis, pasiekiantis virš 80 procentų mP@K įvertį su rparis6k duomenimis, vertinant pagal E protokolą (žr. 1 lentelėje esančioje priede nr. 4).
- Tarp visų modelių, išskyrus mokytų su SphereFace, yra bendri aukšti (virš 60-80 procentų) gldv2_tiny rezultatai. Iš to matyti, kad modeliai kiek prisitaikė prie mokymo žymių, kadangi šiame rinkinyje naudojami tų pačių žymių vietų paveiksliukai kaip ir mokymo metu.

Rezultatai

Atlikto tyrimo metu buvo nagrinėtos 6 nuostolių funkcijos: Trejetų, SupCon, A-Softmax, CosFace, ArcFace bei SubcenterArcFace. Atlikti eksperimentai nagrinėjantys modelio gylio bei duomenų atrinkimo panaudojimo poveikį. Gauti tokie tyrimo rezultatai:

- Atlikta literatūros apžvalga parodė, kad nuostolių funkcijas galima suskirstyti į dvi grupes – vietiniais ir globaliais palyginimais pagrįstas nuostolių funkcijas.
- Geriausi rezultatai, nenaudojant porų atrinkimo, pasiekti CosFace nuostolių funkcija apmokytu ResNet-50 gylio modeliu. Rezultatai labai nežymiai skiriasi nuo ta pačia funkcija apmokyto ResNet-101 gylio modelio.
- Pats geriausias rezultatas gaunamas naudojant Msim porų atrinkimo metodą kartu su SphereFace nuostolių funkcija apmokytu ResNet-101 gylio modeliu.
- Tarp lokaliais palyginimais grįstų funkcijų geriausią rezultatą pasiekia SupCon nuostolių funkciją naudojant apmokytas ResNet-50 gylio tinklas.
- Naudojant porų atrinkimą bendrai geresnius rezultatus pasiekia mažesnio gylio modeliai (ResNet-18 gylio). Nenaudojant porų atrinkimo geriausius rezultatus bendru atveju pasiekia kiek gilesni tinklai (ResNet-50).
- Visi modeliai, apmokyti naudojant gldv2_mokymo duomenų rinkinį nepasiekia HOW metodu gaunamų rezultatų, tačiau rodo geresnius rezultatus negu vaizdų atpažinimui neadaptuoti ResNet tinklai.

Išvados

Atsižvelgiant į atliktą tyrimą ir gautus rezultatus galima priimti šias išvadas:

1. Globaliais palyginimais pagrįstos nuostolių funkcijos (A-Softmax, ArcFace, SubcenterArcFace bei CosFace) pasiekia geresnius rezultatus nei lokaliais palyginimais pagrįstos funkcijos (Trejetų, SupCon) sprendžiant įžymių vietų atpažinimo bei vaizdų atrinkimo uždavinį.
2. Bendru atveju ResNet-101 gylio modeliai nepasiekė geresnių rezultatų. Taip galėjo nutikti dėl mokymo duomenų trūkumo (gldv2_tiny rezultatai rodo gilesnių modelių pritaikymą prie mokymo duomenų).
3. Naudojant porų išgavimą galima pagerinti globaliais palyginimais pagrįstų nuostolio funkcijų rezultatus. Šis metodas padeda pasiekti geresnių rezultatų naudojant mažesnius tinklus, taip sutaupant mokymui reikalingo laiko ir skaičiavimo resursų kiekį.

Šaltiniai

- [DGL⁺20] J. Deng, J. Guo, T. Liu, M. Gong, S. Zafeiriou. Sub-center arcface: Boosting face recognition by large-scale noisy web faces. Iš: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*. Springer, 2020, p. 741–757.
- [DGX⁺19] J. Deng, J. Guo, N. Xue, S. Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. Iš: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, p. 4690–4699.
- [DTP⁺21] M. Douze, G. Tolias, E. Pizzi, Z. Papakipos ir kiti. The 2021 image similarity dataset and challenge. *arXiv preprint arXiv:2106.09672*. 2021.
- [HFW⁺20] K. He, H. Fan, Y. Wu, S. Xie, R. Girshick. Momentum contrast for unsupervised visual representation learning. Iš: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, p. 9729–9738.
- [HZR⁺16] K. He, X. Zhang, S. Ren, J. Sun. Deep residual learning for image recognition. Iš: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, p. 770–778.
- [YOS⁺20] S. Yokoo, K. Ozaki, E. Simo-Serra, S. Iizuka. Two-stage discriminative re-ranking for large-scale landmark retrieval. Iš: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. 2020, p. 1012–1013.
- [KTW⁺20] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, D. Krishnan. Supervised contrastive learning. *Advances in neural information processing systems*. 2020, tomas 33, p. 18661–18673.
- [LWY⁺17] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, L. Song. Sphereface: Deep hypersphere embedding for face recognition. Iš: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, p. 212–220.
- [RIT⁺18] F. Radenović, A. Iscen, G. Tolias, Y. Avrithis, O. Chum. Revisiting oxford and paris: Large-scale image retrieval benchmarking. Iš: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, p. 5706–5715.
- [RTC18] F. Radenović, G. Tolias, O. Chum. Fine-tuning CNN image retrieval with no human annotation. *IEEE transactions on pattern analysis and machine intelligence*. 2018, tomas 41, numeris 7, p. 1655–1668.
- [Sar21] I. H. Sarker. Machine learning: Algorithms, real-world applications and research directions. *SN computer science*. 2021, tomas 2, numeris 3, p. 160.
- [SKP15] F. Schroff, D. Kalenichenko, J. Philbin. FaceNet: A unified embedding for face recognition and clustering. Iš: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, p. 815–823. Prieiga per internetą: <https://doi.org/10.1109/CVPR.2015.7298682>.

- [Soh16] K. Sohn. Improved deep metric learning with multi-class n-pair loss objective. *Advances in neural information processing systems*. 2016, tomas 29.
- [WAC⁺20] T. Weyand, A. Araujo, B. Cao, J. Sim. Google Landmarks Dataset v2 - A Large-Scale Benchmark for Instance-Level Recognition and Retrieval. Iš: *Proc. CVPR*. 2020.
- [WHH⁺19] X. Wang, X. Han, W. Huang, D. Dong, M. R. Scott. Multi-similarity loss with general pair weighting for deep metric learning. Iš: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, p. 5022–5030.
- [WMZ⁺20] Q. Wang, Y. Ma, K. Zhao, Y. Tian. A comprehensive survey of loss functions in machine learning. *Annals of Data Science*. 2020, p. 1–26.
- [WWZ⁺18] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, W. Liu. Cosface: Large margin cosine loss for deep face recognition. Iš: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, p. 5265–5274.
- [WXY⁺18] Z. Wu, Y. Xiong, S. X. Yu, D. Lin. Unsupervised feature learning via non-parametric instance discrimination. Iš: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, p. 3733–3742.
- [XSP20] H. Xuan, A. Stylianou, R. Pless. Improved embeddings with easy positive triplet mining. Iš: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2020, p. 2474–2482.

Sąvokų apibrėžimai

- Požymių žemėlapiai – požymių žemėlapiai tai tarpiniai skaičiavimai (paslėptų sluoksnių reikšmės), kurias tyrimuose dažnai naudoja kaip žemo/aukšto lygio požymius, t.y. kurie gerai užkoduoja nagrinėjamus duomenis.
- Tenzorius – giliojo mokymosi metodų kontekste tai yra n -matė matrica, duomenų struktūra sauganti paveiksluko pikselių reikšmes. Naudojant PyTorch biblioteką tenzoriaus, reprezentuojančio RGB formato paveiksluką, struktūra yra pavidalo $[K, A, P]$ – [kanalų skaičius, paveiksluko aukštis, paveiksluko plotis].
- Aktyvacijos žemėlapis (dar vadinamas požymių žemėlapiu) – dvimatė matrica, gaunama pritaikius konvoliucijos sluoksnių filtrus.
- Daugara – topologinė erdvė, kurios kiekvieno taško aplinka yra artima realiųjų skaičių n -matei erdvei, kurioje yra apibrėžta skaliarinė sandauga (Euklidinei erdvei).
- Hiperparametrai – tai tokie kintamieji, kurie nusako modelio konfigūraciją ir yra fiksuoti ar parinkti. Pavyzdžiui mokymo žingsnis, epochų skaičius.

Priedai

Priedas nr. 1

Nuostolio funkcijų pritaikymo palyginimas – rparis6k mP@K rezultatai

6 lentelė. Galutiniai mP@K rezultatai kai $K = [1, 5, 10]$ su rparis6k duomenų rinkiniu. Rezultatai pateikiami trims vertinimo protokolams - E, M, H.

Modelis	rparis6k E %			rparis6 M %			rparis6 H %		
	mP@1	mP@5	mP@10	mP@1	mP@5	mP@10	mP@1	mP@5	mP@10
R101-GeM-Trejetų	58,57	54,95	52,24	61,43	57,71	55,43	28,57	19,71	16,29
R101-GeM-SupCon	52,86	48,00	46,14	55,71	53,14	51,43	40,00	30,29	26,86
R101-GeM-ArcFace	57,14	58,00	54,86	60,00	60,86	58,57	41,43	32,57	29,43
R101-GeM-SubcArcFace	35,71	33,14	30,43	42,86	42,86	36,57	25,71	22,86	18,29
R101-GeM-CosFace	68,57	62,86	60,00	75,71	67,71	64,57	58,57	45,43	38,14
R101-GeM-SphereFace	21,43	13,43	11,43	27,14	19,71	15,29	10,00	6,86	5,86
R101-Identity	32,86	36,00	35,49	38,57	43,14	43,00	31,43	26,29	23,71
R50-GeM-Trejetų	55,71	54,00	49,86	58,57	57,14	53,57	32,86	23,71	18,71
R50-GeM-SupCon	68,57	64,43	59,43	70,00	68,00	63,57	47,14	36,57	30,86
R50-GeM-ArcFace	54,29	54,29	52,71	60,00	56,86	55,71	37,14	30,29	27,86
R50-GeM-SubcArcface	67,14	60,00	56,43	71,43	64,86	60,86	40,00	33,71	30,29
R50-GeM-CosFace	68,57	63,71	61,57	71,43	66,00	63,71	50,00	39,14	33,86
R50-GeM-SphereFace	14,29	14,00	12,14	14,29	17,43	16,14	< 0,00	4,86	5,57
R50-Identity	22,86	22,00	21,33	32,86	25,86	25,86	18,57	15,71	13,14
R18-GeM-Trejetų	14,29	10,86	10,71	17,14	16,00	15,57	5,71	6,29	6,57
R18-GeM-SupCon	52,86	40,57	35,86	61,43	49,14	43,86	34,29	22,29	19,43
R18-GeM-ArcFace	62,86	61,71	60,71	64,29	65,14	64,57	50,00	44,29	40,86
R18-GeM-SubcArcFace	67,14	62,29	60,71	68,57	64,29	63,14	42,86	36,57	33,86
R18-GeM-CosFace	75,71	66,57	62,43	77,14	70,29	65,57	54,29	39,71	33,71
R18-GeM-SphereFace	17,14	10,57	9,71	20,00	12,57	12,29	2,86	4,00	4,43
R18-Identity	17,14	23,43	26,00	24,29	28,29	30,71	18,57	16,57	13,86

Priedas nr. 2

Nuostolio funkcijų pritaikymo palyginimas – gldv2_tiny mAP, MRR bei P@1 rezultatai

7 lentelė. mAP, MRR bei P@1 rezultatai su gldv_tiny duomenų rinkiniu.

Modelis	gldv2_tiny		
	mAP %	MRR %	P@1 %
R101-GeM-Trejetų	0,23	0,44	0,33
R101-GeM-SupCon	13,48	23,25	20,27
R101-GeM-ArcFace	70,92	82,25	80,77
R101-GeM-SubcenterArcFace	31,88	46,04	41,64
R101-GeM-CosFace	69,93	82,53	80,65
R101-GeM-SphereFace	1,71	2,86	2,36
R101-Identity	15,72	26,57	23,88
R50-GeM-Trejetų	0,24	0,45	0,32
R50-GeM-SupCon	15,08	25,74	22,56
R50-GeM-ArcFace	69,97	81,73	80,06
R50-GeM-SubcenterArcface	69,97	81,77	80,06
R50-GeM-CosFace	67,55	80,50	78,53
R50-GeM-SphereFace	2,49	4,29	3,54
R50-Identity	15,24	25,52	22,89
R18-GeM-Trejetų	0,31	0,59	0,46
R18-GeM-SupCon	7,02	12,66	10,61
R18-GeM-ArcFace	42,44	57,02	54,40
R18-GeM-SubcenterArcFace	44,77	59,53	56,79
R18-GeM-CosFace	41,40	57,28	54,27
R18-GeM-SphereFace	1,84	3,11	2,38
R18-Identity	13,00	22,71	20,31

Priedas nr. 3

Duomenų išgavimo pritaikymas – roxford5k bei rparis6k mAP rezultatai

8 lentelė. mAP įverčiai su roxford5k bei rparis6k duomenų rinkiniais, kai yra naudojamas porų išgavimas mokymo metu. Rezultatai pateikiami trims vertinimo protokolams – E, M, H.

Modelis	roxford5k mAP %			rparis6k mAP %		
	E	M	H	E	M	H
Msim-R101-GeM-ArcFace	9,31	8,76	2,72	37,20	31,29	12,98
Msim-R101-GeM-SubcenterArcFace	15,97	13,97	4,23	40,96	34,73	16,59
Msim-R101-GeM-CosFace	14,50	11,05	1,82	28,72	25,39	11,97
Msim-R101-GeM-SphereFace	27,26	18,34	3,57	56,56	42,62	17,95
R101-Identity	4,62	5,70	2,12	23,80	24,68	13,20
Msim-R50-GeM-ArcFace	18,18	15,87	5,56	40,03	35,20	17,72
Msim-R50-GeM-SubcenterArcFace	19,09	17,05	7,36	46,53	38,28	17,55
Msim-R50-GeM-CosFace	22,09	18,40	9,31	50,38	39,58	17,43
Msim-R50-GeM-SphereFace	27,30	18,35	3,26	57,37	42,79	17,91
R50-Identity	3,62	4,88	2,12	16,12	17,15	7,98
Msim-R18-GeM-ArcFace	23,81	19,65	9,90	48,69	37,58	15,15
Msim-R18-GeM-SubcenterArcFace	22,96	17,31	5,12	43,77	34,84	15,82
Msim-R18-GeM-CosFace	23,74	19,44	8,59	48,45	37,47	17,05
Msim-R18-GeM-SphereFace	20,75	15,80	3,29	60,48	46,22	22,27
R18-Identity	3,21	4,01	1,36	12,72	13,66	6,47

Pastaba: be konteksto gali pasirodyti, kad geriausius rezultatus ResNet-18 gylio modeliuose pasiekia ne SphereFace, bet CosFace nuostolio funkcija, tačiau SphereFace rezultatai pralenkia CosFace mAP įverčius oxford5k bei paris6k rinkiniuose (žr. 5 lentelėje).

Priedas nr. 4

Duomenų išgavimo pritaikymas – roxford5k bei rparis6k mP@K rezultatai

9 lentelė. mP@K rezultatai roxford5k duomenų rinkiniui, kai yra naudojamas porų išgavimas mokymo metu. Rezultatai pateikiami trimis rangams – (1,5,10) bei trimis protokolams (E, M bei H).

Modelis	roxford5 E %			roxford5 M %			roxford5k H %		
	mP@1	mP@5	mP@10	mP@1	mP@5	mP@10	mP@1	mP@5	mP@10
Msim-R101-GeM-ArcFace	17,65	16,67	15,20	18,57	18,14	16,71	2,86	3,48	3,76
Msim-R101-GeM-SubcArcFace	25,00	22,79	20,44	32,86	22,76	19,62	10,00	6,10	5,24
Msim-R101-GeM-CosFace	25,00	21,18	20,18	27,14	20,86	18,57	5,71	4,86	4,00
Msim-R101-GeM-SphereFace	45,59	38,87	32,99	47,14	36,57	31,14	15,71	10,57	7,71
R101-Identity	4,41	3,24	3,38	4,29	4,86	4,86	1,43	2,57	1,86
Msim-R50-GeM-ArcFace	35,29	28,60	26,25	38,57	29,81	27,13	10,00	9,52	8,56
Msim-R50-GeM-SubcArcFace	23,53	21,25	22,43	27,14	23,74	23,65	8,57	11,14	9,04
Msim-R50-GeM-CosFace	29,41	28,24	26,77	31,43	29,14	28,29	12,86	12,57	10,86
Msim-R50-GeM-SphereFace	42,65	35,44	32,75	45,71	33,14	29,14	12,86	7,71	8,29
R50-Identity	<0,00	1,18	1,62	<0,00	7,43	7,86	<0,00	6,86	6,86
Msim-R18-GeM-ArcFace	38,24	32,06	29,52	41,43	33,14	30,14	20,00	14,29	11,43
Msim-R18-GeM-SubcArcFace	38,24	32,35	29,82	38,57	32,00	28,57	8,57	8,93	7,94
Msim-R18-GeM-CosFace	33,82	31,62	29,41	35,71	33,14	30,57	15,71	13,14	12,14
R18-GeM-SphereFace	38,24	30,34	27,94	41,43	31,14	26,71	15,71	7,14	5,43
R18-Identity	<0,00	<0,00	0,59	<0,00	1,43	1,71	<0,00	1,43	1,43

10 lentelė. mP@K rezultatai rparis6k duomenų rinkiniui, naudojant porų išgavimą mokymo metu.

Modelis	rparis6k E %			rparis6k M %			rparis6k H %		
	mP@1	mP@5	mP@10	mP@1	mP@5	mP@10	mP@1	mP@5	mP@10
Msim-R101-GeM-ArcFace	65,71	60,00	55,29	68,57	63,14	59,29	35,71	33,14	30,43
Msim-R101-GeM-SubcArcFace	74,29	68,86	66,00	77,14	72,00	69,57	51,43	44,57	38,00
Msim-R101-GeM-CosFace	62,86	59,14	52,71	70,00	64,00	59,57	45,71	38,86	34,00
Msim-R101-GeM-SphereFace	90,00	85,71	84,14	92,86	88,57	85,57	71,43	57,14	48,00
R101-Identity	32,86	36,00	35,49	38,57	43,14	43,00	31,43	26,29	23,71
Msim-R50-GeM-ArcFace	71,43	62,86	60,86	75,71	72,57	67,71	45,71	44,86	40,00
Msim-R50-GeM-SubcArcFace	77,14	72,00	69,29	78,57	73,71	71,71	55,71	45,71	41,57
Msim-R50-GeM-CosFace	82,86	74,86	72,86	85,71	77,43	74,71	51,43	42,57	36,57
Msim-R50-GeM-SphereFace	88,57	84,86	82,86	91,43	88,00	85,00	75,71	61,14	54,14
R50-Identity	22,86	22,00	21,33	32,86	25,86	25,86	18,57	15,71	13,14
Msim-R18-GeM-ArcFace	81,43	73,14	72,14	84,29	76,57	76,29	51,43	42,29	39,43
Msim-R18-GeM-SubcArcFace	78,57	71,24	66,67	82,86	74,00	70,29	52,86	46,57	42,14
Msim-R18-GeM-CosFace	84,29	74,00	70,71	84,29	76,86	72,57	55,71	48,86	44,00
Msim-R18-GeM-SphereFace	91,43	87,14	83,71	92,86	90,86	87,29	72,86	68,57	60,29
R18-Identity	17,14	23,43	26,00	24,29	28,29	30,71	18,57	16,57	13,86