Lietuvos matematikos rinkinys
Lietuvos matematikų draugijos darbai, ser. B
63 tomas, 2022, 1–13
https://doi.org/10.15388/LMR.2022.29753

Vilnius University Press

Nekilnojamojo turto vertės nustatymas pasitelkiant mašininio mokymosi technikas

Simonas Adomavičius

Matematikos ir gamtos mokslų fakultetas, Kauno technologijos universitetas
Studentų g. 4, Kaunas, Lietuva
El. paštas: adomavicius.simonas@gmail.com

Įteiktas 2022 birželio 28; publikuotas 2022 gruodžio 10

Santrauka. Šiame darbe yra nagrinėjami dirbtinio intelekto metodai, siekiant atlikti tikslesnę Vilniaus mieste ir rajone parduodamų butų vertę. Darbe yra naudojama viešai prieinama informacija apie parduodamus butus iš Aruodas.lt kuri yra surenkama automatizuotu būdu. Informacija kuri yra renkama susideda iš tekstinės – parduodamo buto skelbimo aprašymas, nuotraukų – skelbime patalpintos nuotraukos, bei bendrinė informacija pateikiama skelbime – kaina, vietovė, buto plotas, buto ypatumai ir kita. NT vertės nustatymo uždaviniuose nuolat pasitaikanti problema yra mažai vertės turinčių objektų pervertinimas ir/ar didelę vertę turinčių objektų nepakankamas vertinimas. Sprendžiant regresijos uždavinius, mes dažnai turime duomenų apie daugumą objektų, tačiau visuomet per mažai itin pigių, bei itin prabangių. Dėl šios priežasties vertinti daugumos objektų vertę yra lengviau, nei pigių ar brangių. Vis dėl to, dėl tobulėjančių dirbtinio intelekto metodų bei informacijai tampant vis lengviau pasiekiamai, mūsų galimybės geriau įvertinti šio tipo būstus tampa vis didesnė. Darbe tikimasi, jog informacija esanti nuotraukose ir tekste leis atlikti geresnę būsto vertės prognozė geriau vertinant tiek pigius, tiek brangius butus. Pirmojoje darbo dalyje atliekama literatūros apžvalga, kitų autorių darbų nagrinėjusių dirbtinio intelekto panaudojimo galimybes būsto vertės prognozavimui. Antroje dalyje aprašomi tyrimo metodai, kurie bus taikomi darbe ir pristatoma informacijos rinkimo strategija. Trečiojoje dalyje yra atliekamas tyrimas, kurio metu iš pradžių yra atliekama požymių inžinerija, o vėliau modelių apmokymas bei optimizavimas. Galiausiai yra pristatomi geriausio modelio su 13.74 MAPE, ir 33,307 RMSE rezultatai, bei pateikiamos išvados.

Raktiniai žodžiai: mašininis mokymasis; regresija; būsto vertės nustatymas; teksto analizė; vaizdų analizė; klasterizavimas; aruodas.lt

AMS: 68T07

Įvadas

Vertės nustatymas bei kainos prognozavimas yra nuo seno sprendžiami uždaviniai. Vieni uždaviniai keliami siekiant numatyti, kaip keisis valiutų, akcijų ar prekių kainos per tam tikrą laiką, kiti – keliami tam, kad pasitelkus atitinkamus požymius apie objektą būtų apskaičiuota galima jo vertė. Nors pirmieji yra laiko eilutės uždaviniai, o antrieji – regresiniai, kurie anksčiau buvo sprendžiami atskirai, dabar, kai darosi vis lengviau surinkti daugiau informacijos bei besivystant naujiems ir pažangesniems metodams – gali būti atlikti kartu siekiant pasiekti geresnius rezultatus. Negana to, atsiranda vis daugiau galimybių įtraukti vis daugiau informacijos į modelių sudarymą bei atsirinkti požymius, geriausiai išsprendžiančius šiuos uždavinius.

Gerai sudaryti modeliai gali būti naudojami prognozuoti įvairią vertę ar kainą, pavyzdžiui, automobilių, nekilnojamojo turto, žaliavų, valiutų ir kt. Tačiau ypač aktualūs yra modeliai skirti NT vertei. Šie modeliai ypač naudingi bankams, kadangi už kiekvienos paskolos egzistuoja įkeistas turtas, o finansinės įstaigos privalo prisiimti riziką klientui bankrutavus ir tapus nemokiam. Bankai, kurie patys vykdo savo kredito rizikos valdymą A-IRB (angl. *Advanced Internal Rating-Based*), privalo stebėti pagal Bazelio (angl. *Basel*) reikalavimus įkeisto turto vertę, jog atitiktų visus rizikos standartus. Taip pat dideli bankai turi klientų ne tik vienoje šalyje, o keliose. Todėl gerai įvairius NT objektus galintis vertinti modelis tampa ypač paklausus. Vienintelė problema, jog modelio paaiškinamumo stoka (dažnai neatsiejama nuo sudėtingesnių modelių) taip pat yra rizika bankui, todėl modelio paaiškinamumas ir galimybė lengvai jį interpretuoti taip pat yra itin svarbūs modelio elementai.

Tikslus NT kainų prognozavimas atneštų naudos ne tik bankams, bet ir kitoms finansinėms įstaigoms, pavyzdžiui, draudimo. Žinant turto vertę galima tikslingai atsidėti kapitalą priimamai rizikai atsverti. Nekilnojamojo turto sektorius taip pat turi daug potencialo, nes tikslus modelis galėtų pakeisti būsto vertintojus ir tapti skaidri ir nešališka alternatyva vertinti NT vertę.

Taigi, modelis gebantis tiksliai prognozuoti NT vertę turi gana platų pritaikomumą, tačiau uždavinio sprendimas nėra paprastas, dėl didelio neužtikrintumo NT vertėje. Būsto vertę įtakoja ne tik ekoniminiai veiksniai tokiai kaip infliacija, kylantys atlyginimai, nedarbo lygis ir k.t, tačiau ir abstraktūs ar ne tokie „pamatuojami“ kriterijai būdingi konkretiems objektams. Pavyzdžiui, identiškas butas tame pačiame name gali kainuoti skirtingai nuo daug faktorių: įrengimo (ar bendros objekto kuriamos atmosferos/auros), parkingo vietos, panoramos vaizdo, ar šiame bute gyveno istorinė asmenybė ar kt. Būtent ši informacija ir yra prieinama teksto ir vaizdo pavidalu. Ne be reikalo, NT brokeriai skiria daug pastangų į kokybiškas nuotraukas, paryškinančias objekto privalumus bei į pirkėjo patiriamą vertę orientuotus aprašymus.

Todėl darbe bus siekiama atsižvelgti į šiuos aspektus regresijos lygtį praturtinant nuotraukomis, bei tekstu. Tai bus atliekama iš Aruodas.lt surinktais duomenimis. Darbe toliau išbandomi įvairūs modeliai ir technologijos, siekiantys kuo tiksliau įvertinti būsto kainą.

1 Literatūros apžvalgos rezultatai

Darbe buvo stengtasi į būsto kainos prognozavimo uždavinį pažiūrėti iš įvairių pusių: kaip regresinį uždavinį, BKI ir kaimynų uždavinius, giliojo mokinimosi uždavinį, dau-

1 lentelė. Litetarūros rezultatai.

Šaltinis	Algoritmas	Rezultatas
[2]	Ensembles of regression trees	MAE = 0.371(0.0075), MSE = 4.346(0.1548)
[14]	Stacked Generalization Regression	RNSKE = 0.16350
[5]	RF+LSTM Ensemble	RMSLE = 0.23847
[13]	NN	R2 = 93, MSE = 0.001
[9]	KNN MC	MDAPE = 0.0831
[4]	Geo-spatial network embeddings (GSNE)	RMSE = 0.181, MAE = 0.125
[6]	MTL	RMSE = 0.184–0.262, MAE = 0.135–0.187
[7]	GBM	R2 = 90.4, RMSE = 0.08903, MSE = 0.00793, MAPE =0.32251
[8]	0.65Lasso+ 0.365Xgb	RMSE = 0.11260
[10]	LSSVM	MAPE = 0.228
[11]	RIPPER	MAE = 0.2488
[12]	Stepwise and tuned SVM,	MSE = 0.0561
[3]	GB+LSTM	MAPE = 0.2403

giaužduotinį uždavinį ar kaip laiko eilutes. Apžvelgtų darbų rezultatai yra pateikiami 1 lentelėje. Svarbu paminėti, jog darbuose skyrėsi modelio vertinimo kriterijai, todėl buvo pateiktos įvairios metrikos.

Galime matyti, jog darbuose buvo naudojami įvairūs modeliai, kurie praktikoje yra gana populiarūs. Taip pat apibendrinus rezultatus, galima pastebėti jog geriausi rezultatai buvo pasiekti naudojant apie būsto skelbimo susijusią lokaciją. Tam dabar yra daug galimybių pasitelkiant Google žemėlapių API: galime ne tik naudodami būsto koordinates gauti satelito nuotraukas, tačiau ir pasiskaičiuoti atstumus iki viešojo transporto stotelių, traukinio stočių. Taip pat aukštus rezultatus pasiekė ir modeliai, kurie įtraukė kiek įmanoma daugiau požymių per teksto ir nuotraukų atpažinimo modelius.

Vienas įdomesnių sprendimų buvo naudoti daugiaužduotinį mokymąsi (MTL), kuris plačiai yra naudojamas Tesla automobiliuose. Pagal šį modelį buvo sukurti įvairūs uždaviniai, kurie sprendė būsto kainas skirtinguose rajonuose, reitingavo vietines mokyklas ir t.t. Apjungiant įvairius modelius būtų galima pagalvoti apie MTL architektūra kuri ne tik bando nustatyti kainą skirtinguose rajonuose, tačiau ir įvertinti būsto būklę iš nuotraukų ar pridėtinę vertę iš skelbimo aprašymo. Taip pat galime įkomponuoti ir „Atviras Vilnius“[2] atvirus duomenis apie darželius, mokyklas ir kt.

Taip pat keliuose darbų buvo pastebėta, jog pašalinus išskirtis – itin aukštos kainos, ar itin žemos kainos būstus – modelio rezultatai pagerėjo. Kitas svarbus dalykas, jog pastebima tendencija, jog vieno modelio, kaip sprendimo – nebepakanka ir yra pradedama stengtis apjungti įvairius modelius į vieną. Šiam tikslui pasiekti yra įvairių architektūrų tačiau tos, kurios buvo pastebėtos apžvalgoje buvo ansambliai, sluoksniavimas (angl. *Stack*), MTL ar kaip daugybinės įvesties neuroninius tinklus (angl. *Multi-Input Neural Network*).

Taip pat pastebėta, jog nėra itin daug darbų, kurie nagrinėti kitokias architektūras nei regresija per mašininį ar gilųjį mokymąsi. Dauguma sprendimų apsiriboja kurdami sudėtingesnius sprendimus bandydami įtraukti vaizdą, tekstą, bei papildo-

[2] https://atviras.vilnius.lt/.

mą informaciją į galutinę regresijos prognozę. Vis dėl to, ir savo darbuose sugebėjo įkomponuoti ir laiko eilutes, įtraukdami makroekonominius veiksnius į modelių prognozavimą. Tačiau kaip matyti per rezultatus, vieno galutinio sprendimo nėra, tačiau kaip jau buvo galima matyti – įvairių sprendimų kombinacija dažnai lemia geresnius sprendimus, nei individualūs sprendimai modelio kompleksiškumo kaina.

Galiausiai, dažniausias NT prognozės sprendimas yra per regresijos uždavinį, todėl darbe taip pat bus naudojamas šis metodas. Taip pat tai yra gana optimalus pasirinkimas atsižvelgiant į itin mažą duomenų imtį.

2 Tyrimo metodai

2.1 Duomenų rinkimas

Duomenų rinkimui iš Aruodas.lt buvo naudoti 2 pagrindiniai Python paketai: BeautifulSoup ir Selenium. Pirmoji biblioteka yra skirta HTML ir XML apdorojimui, kurios dėka visas HTML turinys esantys puslapyje gali būti skaitomas ir randamas naudojant atitinkamus paieškos metodus. Vis dėl to, tam tikri puslapiai gali būti sukurti taip jog naudotų dinaminį tekstą naudojant JavaScript. Tokio teksto su anksčiau minėta biblioteka rasti nepavyktų, kadangi pastaroji nuskaito tik HTML ir XML informacija. Dėl šios priežasties buvo pasitelkta kita biblioteka – Selenium.

Dėl duomenų rinkimo iš interneto, svarbu prisiminti GDPR ir Intelektualinės nuosavybės įstatymus. Pagrindiniai principai kurių derėtų laikytis:

1. Siekti kuo labiau sumažinti naštą interneto svetainės savininkams;
2. Patenkinti svetainių savininkų pateiktus prašymus dėl duomenų rinkimo, aprašyto „robots.txt“ faile;
3. Saugoti visus asmeninius duomenis visose statistikos ir tyrimų rezultatuose;
4. Taikyti mokslinius principus rengiant statistiką ir tyrimus, pagrįstus nuasmenintais/užšifruotais duomenimis;
5. Laikytis visų galiojančių teisės aktų ir stebėti besikeičiančią teisinę situaciją.

Atsižvelgus į šiuos punktus duomenys buvo surinkti skaidriai ir etiškai.

2.2 Vaizdo ir teksto vektorizavimas

Siekiant įtraukti skelbimų aprašymus į regresijos lygtį, tekstą transformuosime į vektorinę formą. Paties metodo esmė labai paprasta ir artima NLP uždaviniams, kuomet yra sudaromas TF-IDF ar BoW žodžių rinkinys, kur vėliau žodžiai/frazės yra susiejami pagal savo panašumą. Tačiau vietoj teksto prognozavimo, žodžiai paversti skaitine išraiška ir susietini panašumu yra naudojamas kaip vektorius modeliams naudojantiems skaitinio tipo kintamuosius.

Žodžių ir frazių pasiskirstymo ir sudėties nustatymui galimi 2 metodai: tęstinis žodžių maišas (angl. *Continuous Bag of Words*) ir n-gramų įterpinių (angl. *The skip gram model*). Pirmasis apima konteksto žodžių nuspėjimą naudojant centrinį žodį, o kitas apima žodžio nuspėjimą naudojant kontekstinius žodžius. Tą pačią idėją galima išplėsti sakiniams ir dokumentams. Word/Sent/Doc2Vec remiasi n – gramų įterpiniais. Ši savybė ir yra šių modelio tipo išskirtinumas lyginant su kitais modeliais.

Kadangi mūsų darbe teksto nebus daug – NT būstų skelbimų aprašymai – savo darbe naudosime sakiniams skirtą modelį – Sent2Vec.

Literatūros apžvalgoje autoriai naudojo konvoliucinius neuroninius tinklus, tam kad galėtų išmokinti modelį atpažinti atitinkamas savybes esančias nuotraukose siejamomis su vienokia ar kitokia kaina. Kadangi darbe NT vertės prognozavimas yra atliekamas regresiniu būdu, kaip ir tekstą – vaizdą taip pat vektorizuosime. Vektorizavimui bus naudojamas iš anksto apmokintas *EfficientNetB0* modelis. EfficientNet yra ypač įspūdingas, nes jame yra automatiškai sugeneruotų modelių klasė su kompromisu tarp parametrų skaičiaus ir tikslumo. Modelių klasės skirstomos nuo B0 iki B7, kur parametrų skaičius atitinkamai 5.3 mln parametrų, o didžiausiame B7 – 66 milijonai.

Yprastai, šis modelis yra naudojamas klasifikacijos uždaviniams atlikti, tačiau nuėmus paskutinį sluoksnį iš architektūros, galime gauti 1280 dydžio vektorių (naudojant EfficientNetB0), kuris atspindi modelio paskaičiuotas ypatybių reikšmes, leidžiančias atlikti galutinę prognozę. Naudojant šį metodą, galėsime vektorizuoti visus paveikslėlius ir įtraukti juos skaitine išraiška į skaičiavimus.

2.3 Kiti naudoti metodai

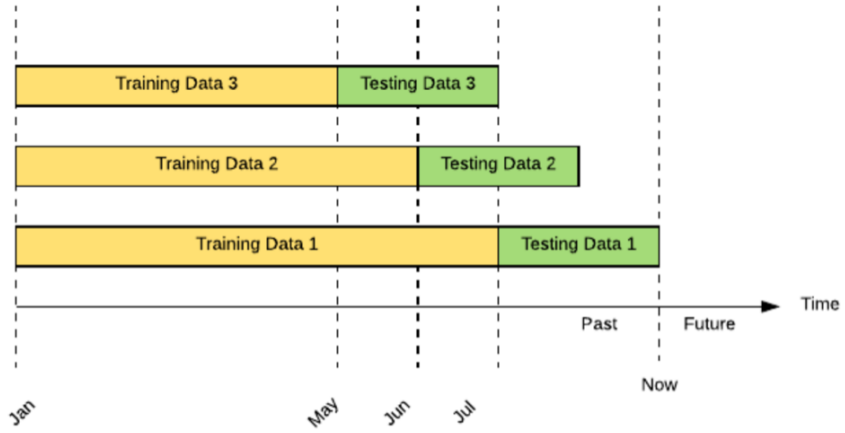
Kadangi darbe bus naudojama regresija, bus taikomi pagrindiniai metodai naudoti literatūros apžvalgoje. Daugiausia šių modelių yra medžių šeimai priklausantys modeliai, tačiau papildomai panaudoti ir tiesiniai modeliai, kurie nors ir nėra tokie galingi kaip pastarieji – vis dėl to yra vertinami versle iki dabar dėl paprasto interpretuojamumo ir paaiškinamumo. Šalia šių dviejų modelių šeimų taip pat bus panaudota artimiausių kaimynų regresija, vienas NN modelis ir įvairių modelių ansambliai. Visi naudoti modeliai pateikiami 2 lentelėje.

Atliekant požymių inžineriją, bus siekiama ne tik atlikti prasmingų naujų požymių kūrimą (pvz.: atstumas iki centro, Vingio parko), tačiau ir dirbtinai išplėsti požymių skaičių atliekant polinomų transformaciją tolydiems kintamiesiems ir dažnių charakteristiką kategoriniams kintamiesiems. Tai atliekama siekiant padėti modeliams greičiau atrasti ryšius tarp požymių.

Kadangi požymiai buvo išplėsti, taip pat yra naudojama Boruta biblioteka, reikšmingų požymių atrankai. Naudojant šį metodą vėliau yra siekiama patikrinti, ar verta turint daug požymių leisti modeliui turinčiam reguliarizacijos mechanizmus savyje pačiam atsirinkti požymius, ar geriau jei tai padarytų kitas metodas.

Hiperparametrų paieškai buvo pasitelkta Optuna [1] biblioteka. Ši biblioteka pritaiko Bajeso teoremą paieškoje, todėl optimalūs hiperparametrai yra randami greičiau. Optuna taip pat atlieka skaičiavimus paraleliai pagal turimus kompiuterio branduolius bei turi gražias ir paprastas rezultatų vizualizacijas.

Modelio mokinimas ir jo kokybės vertinimas bus skaičiuojamas naudojant kryžminę patikrą. Ši kryžminė patikra bus ypatinga tuo, jog kiekvieno mėnesio turimi duomenys, bus naudojami prognozuoti sekančio mėnesio būstų kainas. Kitas svarbus dalykas, jog prie kiekvieno mėnesio bus pridedami visi prieš tai buvę mėnesiai nuo pat duomenų rinkimo pradžios – 2022 Sausio mėnesio. Ši besiplečiančio lango strategija (angl. *expanding window*) (1 paveikslėlis) buvo priimta dėl mažos duomenų imties – 4354 įrašų, o sekančio mėnesio prognozavimas naudingas tuo jog leidžia netiesiogiai atsižvelgti į makroekonominius kainos augimo veiksnius.



1 pav. Slenkančio lango architektūra.

3 Mokslinis tyrimas

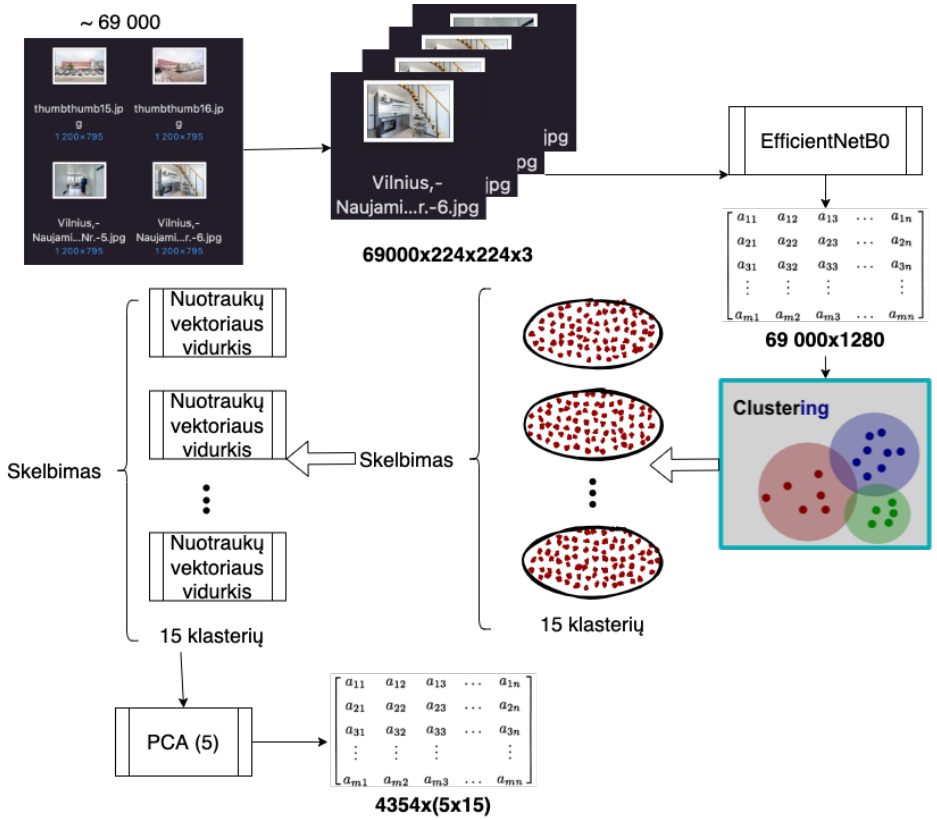
3.1 Požymių inžinerija

Teksto vektorizavimas darbe buvo atliktas itin paprastai, kadangi visą darbą atliko Sent2Vec[3] modelis. Modelio implementacija Python kode buvo atlikta patogiai, todėl užteko parinkti tik tinkamus parametrus, bei tekstą. Viskas kas buvo atlikta tai teksto valymas ir tvarkymas. Šis žingsnis leido sukurti 16 požymių per skelbimą. Žemiau pateikiami naudoti modelio parametrai:

1. Minimalus žodžių pasikartojimų skaičius [minCount] = 8;
2. Rezultato dimensijos [dim] = 16;
3. Epochų skaičius [epoch] = 9;
4. Mokymosi dydis (angl. “learning rate”) [lr] = 0.2;
5. Maksimalus ngram žodžių kiekis [wordNGrams] = 2;
6. Nuostolio funkcija [loss] = ns;
7. Atrinktų negatyvų skaičius [neg] = 10;
8. Gijų skaičius naudojamas skaičiavimams kompiuteryje [thread] = 2;
9. Atrankos slenkstis (angl. “sampling threshold”) [t] = 0.000005;
10. Išmetamų ngramų skaičius mokinant modelį [dropoutk] = 4;
11. Minimalus Y(label) atvejų skaičius [minCountLabel] = 20;
12. “Hash bucket” skaičius duomenų rinkiniui [bucket] = 4000000;
13. Maksimalus duomenų rinkinio dydis sakiniais [maxVocabSize] = 750000;
14. Tarpinių rezultatų išsaugojimo dažnumas [numCheckPoints] = 10.

Vaizdo vektorizacijai buvo pasirinkta strategija pavaizduota 2 paveikslėlyje. Iš pat pradžių nuotraukų dydis buvo pakeistas į dydį priimtiną EfficienetNetB0 – 224×224.

[3] https://github.com/epfml/sent2vec.



2 pav. Klasterizavimo architektūra.

Praleidus visas nuotraukas pro modelį, kiekvienai nuotraukai buvo priskirtas 1280 dydžio vektorius. Iš viso turime (69000, 1280) dydžio matricą.

Toliau atliekame klasterizavimą naudojant centroidų giminės modelį KMeans. Su šia operacija siekiame suskirstyti panašius paveikslėlius į klasterius. Naudojant alkūnės metodą yra nustatomas optimalus klasterių skaičius – 15. Tuomet kiekvienai nuotraukai apskaičiuojame ir priskiriame jai atitinkamą klasterį. Atliekant klasterizavimą, laikinai atliekamas dimensijų mažinimas išlaikant 95% sklaidos, siekiant pagreitinti skaičiavimus, tačiau tolimesniuose žingsniuose yra naudojamas pradinis 1280 dydžio vektorius per nuotrauką.

Galiausiai skelbimo lygmenyje, paskaičiuojame koks yra kiekvieno iš 15 klasterių nuotraukų vektoriaus vidurkis (jei skelbime kažkuris klasteris yra tuščias, jį žymime 0). Šiame etape turime 15×1280 požymių per skelbimą, todėl atliekamas PCA(5) siekiant suspausti požymių dimensijas per klasterį. Galiausiai su šiuo veiksmu liekame su 15×5 požymiais per skelbimą.

Be šių veiksmų, taip pat buvo atlikti standartiniai mašininio mokymosi veiksmai, tokie kaip: išskirčių valymas, tuščių reiškmių tvarkymas, naujų požymių kūrimas atsižvelgiant į žvalgomosios analizės rezultatus, bei literatūros apžvalgoje akcentuotus požymius ir t.t.

3.2 Hiperparametrų paieška

Visų pirma, prieš pradedant hiperparametrų paiešką, yra sudaroma optimali grandinė (angl. *Pipeline*) transformacijų, būdingo tiesiniams ir medžių modeliams. Bendrai, šios transformacijos apima: kategorinių konvertavimą į fiktyvius kintamuosius, standartizavimą, bei požymių išplėtimą naudojant polinomų transformacijas tolydiems kintamiesiems, bei dažnių charakteristiką kategoriniams kintamiesiems prieš transformaciją į fiktyviuosius. Papildomai, tiesinių modelių grandinė turi papildomą transformaciją, kuri pasirūpina, jog visi tolydieji kintamieji turėtų normalųjį skirstinį. Sprendimas sukurti tokias grandines atėjo tikrinant įvairias transformacijų kombinacijas naudojant tinklelio paiešką.

Toliau atliekama hiperparametrų paieška, kurios metu papildomai ieškoma ar visiems tiesiniams modeliams, verta atlikti regresanto transformaciją paverčiant jo skirstinį į Gauso ir ar reikšmingų požymių atranka padeda pagerinti modelio rezultatus. Požymių atranka buvo atlikta naudojant 2 metodais, vienas iš jų pristatytas ankstesnėje dalyje – Boruta, kuris atrado 61 požymį. Antras metodas buvo naudojant Lasso regresiją ir atrinkus didžiausius koeficientus (po duomenų normalizavimo), siekiant gauti 100–200 požymių. Pirmuoju metodu paieška užtruko 102 minutes, o antruoju – 2.

Galiausiai yra atliekama visų parametrų paieška naudojant Optuna ir pateikiami paieškos rezultatai, iteracijų skaičius ir trukmė 2 lentelėje.

2 lentelė. Optuna rezultatai.

Model	Optuna rounds	RMSE	Features	Inverse	Total user time (min)	Total CPU time (min)	AVG CPU time (s)
Lasso	100	40589	ALL	TRUE	7	29	0,28
Ridge	100	41082	ALL	TRUE	6	25	0,25
ElasticNet	300	43474	ALL	FALSE	16	62	0,2
SVR	391	43728	BORUTA	TRUE	40	160	0,4
SDGRegressor	250	49432	BORUTA	TRUE	12	46	0,2
BayesianRidge	97	42144	ALL	TRUE	82	328	3,38
MLPRegressor	8	46840	LASSO	N/A	402	N/A	N/A
RandomForest	334	40091	BORUTA	N/A	68	270	0,8
GradientBoostingRegressor	71	34547	ALL	N/A	75	299	4,2
XGBoost	56	35148	ALL	N/A	158	631	11,27
Catboost	8	37755	ALL	N/A	88	353	224,16
AdaBoostRegressor	100	48874	BORUTA	N/A	39	156	1,55
BaggingRegressor	100	39532	BORUTA	N/A	37	147	1,47
ExtraTreesRegressor	122	39209	BORUTA	N/A	17	67	0,55
HistGradientBoostingRegressor	39	35940	ALL	N/A	71	282	2,57
LightGBM	250	37450	ALL	N/A	30	120	0,5

3.3 Rezultatai

Atradus geriausias modelio parametrų kombinacijas, buvo paskaičiuoti kiekvieno modelio rezultatai. Tuomet buvo atrinkti TOP 5 geriausi modeliai pateikiami 3 lentelėje ir panaudoti ansamblio architektūroje.

Toliau yra išbandomos šios ansamblių technikos iš aukščiau pateiktų modelių:

1. Visų modelių prognozių rezultatus su meta modeliu – paprasta tiesine regresija (angl. *Stacking*);
2. Visų modelių prognozėms paprasčiausiai paskaičiuoti vidurkį;

3 lentelė. TOP 5 geriausių modelių rezultatai.

Model name	R2 score	MAE	MAPE	MSE	RMSE	Time
Gradient Boosting Regressor	0.86	21,850.95	15.27	1,193,510,824.54	34,547.23	59.03
XGBoost	0.86	22,256.48	15.45	1,235,424,544.62	35,148.61	184.35
Histogram-based Gradient Boosting Regression Tree	0.85	21,994.76	14.59	1,291,711,487.63	35,940.39	202.42
LightGBM	0.84	23,949.97	16.64	1,408,119,259.42	37,524.92	8.88
Catboost	0.84	23,385.94	16.11	1,433,786,874.95	37,865.38	850.42

4 lentelė. TOP 5 geriausių modelių ansambliai.

Model name	R2 Score	MAE	MAPE	MSE	RMSE	Time
Stacking Models	0.87	20,695.76	13.99	1,125,785,632.75	33,552.73	1,907.69
Averaging Models	0.87	20,663.98	14.02	1,156,017,310.87	34,000.25	359.63
Ensemble Prediction	0.85	22,203.91	14.69	1,335,156,497.66	36,539.79	340.07
Residual Model	0.87	20,224.45	13.74	1,109,365,262.59	33,307.14	439.82

3. Atlikti visų modelių balsavimą, kur kiekvieno modelio balso svoris yra nustatomas taip, jog paklaida būtų mažiausia, o visų modelių bendras balsas būtų lygus 1;
4. Parinkti geriausią ansamblio architektūrą ir iš jo liekanų sukurti antrą modelį, kuris pakoreguotų pirmojo paklaidas, kaip dar vieną mėginimą pagerinti modelio rezultatus.

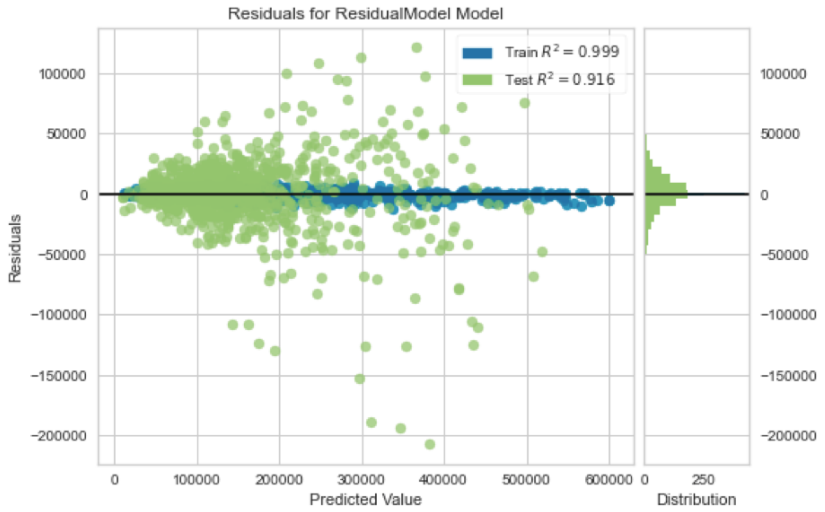
Šių ansamblių rezultatai yra pateikiami 4 lentelėje.

Taigi, geriausias modelis ir šio projekto finalinis modelis yra liekanų modelis sudarytas iš TOP 5 geriausių individualių modelių prognozės vidurkio, bei ant viršaus sukurto papildomo liekanų modelio naudojantis geriausio individualaus modelio architektūrą – gradientinio pastiprinimo modelis su optimaliais parametrais. Šis modelis pasiekė 13.74 MAPE ir 33,307 RMSE įverčius.

Liekanų grafike (3 pav.), galime pastebėti, jog modelis puikiai įsimena mokymo imtį, tačiau nepersimoko ir gali tinkamai prognozuoti objektus kurių nėra matęs. Taip pat galime matyti, jog modelis panašu jog daugiau objektų yra pakankamai nesuvertinęs (didelės paklaidos iki 200 tūkst. Eur.), tačiau ne tiek daug pervertina (paklaidos iki 130 tūkst. Eur.). Vis dėl to, vertinant testavimo imties paklaidų skirstinį, galime įžvelgti jog skirstinys yra pasislinkęs į kairę nuo nulio, kas reiškia, jog testavimo imtyje modelis linkęs bendrai pervertinti būstus, nei nesuvertinti pakankamai. Taip pat galime matyti, jog mokymosi imtyje buvo suvertinti ir vidutinės vertės būstai ir brangesni, tačiau testavimo duomenyse sunkiau sekėsi įvairios vertės būstams, ne tik itin prabangiems.

Sprendžiant ar modelis galėtų būti naudojamas versle, svarbu atsižvelgti į jo kokybę portfelio lygiu – dažnais atvejais mes norėtume jog >80% viso portfelio paklaidos nuo realios vertės būtų $(-15; 15)$ procentų intervale. Vis dėl to, darbo modelio rezultatai yra kiek prastesni.

Pagal turimą liekanų skirstinį skirstinį (5 lentelė), modelio prognozuojamoji „galia" yra per menka, jog būtų sėkmingai naudojama versle, juolab kad modelio architektūra ir pateikiamų rezultatų paaiškinamumas yra sudėtingas.



3 pav. Geriausio modelio liekanos mokymosi ir treniravimosi imtyse.

5 lentelė. Geriausio modelio paklaidų pasiskirstymas.

Tier	Percentage
$(-15; 15)$	0.68
$(-25; 20)$	0.15
Other	0.17

Išvados

1. Literatūros apžvalgoje buvo rasti įvairūs metodai, siekiantys atlikti tikslesnę NT kainos prognozę, tačiau esminiai dalykai buvo šie: vietovės informacijos išnaudojimas skaitine ar paveikslėlio išraiška, būsto nuotraukos, duomenų praturtinimas įvairia papildoma informacija. Buvo naudoti įvairūs modeliai, tačiau modelių ansambliams pavykdavo pasiekti geriausius rezultatus.
2. Informacijos iš Aruodas.lt rinkimas buvo atliktas naudojant Selenium įrankį. Atliekant informacijos rinkimą, svarbu atsižvelgti į svetainės, iš kurios bus renkama informacija resursus ir pagal tai sukurti etišką duomenų rinkimo strategiją. Verta paminėti, jog informacijos rinkimas Europoje vis dar yra "pilkoji zona", todėl svarbu žinoti, ką ir kaip galima rinkti.
3. Teksto įtraukimui į regresijos modelį buvo naudotas Sent2Vec modelis, kurį naudojant sutvarkytas tekstas buvo transformuotas į 16 dydžio vektorių. Atlikus požymių atranką paaiškėjo, jog tekstas yra vienas iš požymių, kuris buvo svarbus atliekant kainos prognozę. Šiuos rezultatus patvirtino požymių atranka naudojant Boruta, bei permutacijos būdu įvertinta požymių svarba iš geriausio modelio. Taip pat, jeigu modelį būtume atskirai apsimokinę ant lietuviško tekstyno - tikėtina jog rezultatai galėtų būti geresni.
4. Vaizdui įtraukimo į regresijos modelį buvo naudotas EfficientNetB0 modelis. Šis mažiausias modelis buvo pasirinktas todėl, jog modelio greitaveika (angl.

inference) buvo vykdoma lokaliai, esant ribotiems kompiuterio resursams. Su šiuo modeliu, vaizdai buvo suspausti į 1280 dydžio vektorių, tuomet naudojant artimiausių kaimynų modelį, buvo suskirstyti į klasterius. Visos nuotraukos tuomet buvo suvidurkintos savo klasteryje, per parduodamo būsto skelbimą. Atlikus požymių atranką paaiškėjo, jog šiuo būdu įtraukti paveikslėliai nebuvo naudingi atliekant prognozes, o permutacijos būdu įvertinus požymių svarbą – šie požymiai turėjo žemiausius (neigiamus) įverčius. Vis dėl to, tai nereiškia, jog nuotraukos nėra reikšmingos. Pasirinktas sprendimas tik parodė, koks sudėtingas yra uždavinys iš nuotraukos išgauti subjektyvius vertę suteikiančius požymius. Todėl toliau plėtojant šią temą, derėtų skirti pakankamai laiko tinkamos architektūros/modelio paieškoms.

5. Šalia modelių parametrų atrankos, taip pat buvo vertinama, ar informatyvių požymių atranka pagerina modelio prognozuojamąją galią, esant optimaliems parametrams. Analizė parodė, jog tai priklauso nuo paties modelio – vieniems modeliams tai ne tik pagreitina greitaveiką, tačiau ir pagerina rezultatus, net ir tuomet kai patys modeliai turi požymių atrankos mechanizmus patys savyje. Kitiems modeliams priešingai, geresni rezultatai buvo pasiekti naudojant visus požymius. Kita vertus, verta paminėti, jog naudojant gamyklinius/bazinius modelio parametrus – požymių atranka gali ne tik padidinti greitaveiką, tačiau ir rezultatus (modelio tikslumą). Dėl šios priežasties, siekiant greitų rezultatų praktikoje, vertėtų pradėti nuo greitų sprendimų, kurie nereikalauja eikvoti išteklių ieškant optimalių parametrų ir požymių kombinacijų, naudojant bazinius parametrus ir atrinktus požymius. Taip pat darbe pastebėta jog regresanto normalumas leidžia pasiekti geresnius rezultatus tiesiniams modeliams.
6. Lyginant atskirus modelius su geriausių modelių ansambliais, pasiteisino literatūroje pastebėta tendencija dėl ansamblio rezultatų pagerinimo. Dauguma ansamblio architektūrų parodė geresnius rezultatus, nei TOP 5 modeliai atskirai, tačiau visų jų rezultatai buvo beveik identiški. Vis dėl to, rezultatai nepagerėjo itin daug, tačiau greitaveikos laikas – prailgėjo pastebimai. Teoriniai/eksperimentiniai modeliai yra puikūs dėl savo rezultatų, tačiau taikant juos produkcijoje, verta atsižvelgti į jų greitaveikos laiką, ypač kai turime milijonus duomenų, bei į architektūros sudėtingumą, kai reikia paaiškinti kodėl yra prognozuojama vienaip, o ne kitaip. Geriausias modelis prognozuojantis būsto vertę buvo modelių ansamblis atliekantis prognozių vidurkio skaičiavimą, bei ant viršaus papildomas liekanų modelis naudojantis geriausio individualaus modelio architektūrą – gradientinio pastiprinimo modelis su optimaliais parametrais. Šis sprendimas sugebėjo pasiekti mažiausias prognozavimo klaidas (MAPE = 13.74, RMSE = 33,307).

Literatūra

[1] T. Akiba, S. Sano, T. Yanase, T. Ohta, M. Koyama. Optuna: A next-generation hyperparameter optimization framework. In *25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019.

[2] A. Baldominos, I. Blanco, A.J. Moreno, R. Iturrarte, Ó. Bernárdez, C. Afonso. Identifying real estate opportunities using machine learning. *Appl. Sci.*, **8**, 2018. https://doi.org/10.3390/app8112321.

[3] J. Bin, S. Tang, Y. Liu, G. Wang, B. Gardiner, Z. Liu, E. Li. Regression model for appraisal of real estate using recurrent neural network and boosting tree. In *International Conference on Computational Intelligence and Applications*, ICCIA, 2018.

[4] S.S. Sarathi Das, M.E. Ali, Y.-F. Li, Y.-B. Kang, T. Sellis. Boosting house price predictions using geo-spatial network embedding. *Data Min. Knowl. Disc.*, **35**:2221–2250, 2020. https://doi.org/10.1007/s10618-021-00789-x.

[5] B. Klaus de Aquino Afonso, L.C. Melo, W.D. Gomes de Oliveira, S.B. da Silva Sousa, L. Berton. *Housing Prices Prediction with a Deep Learning and RandomForest Ensemble.* 2019.

[6] G. Gao, Z. Bao, J. Cao, A.K. Qin, T. Sellis, Fellow, IEEE, Z. Wu. Location-centered house price prediction: a multi-task learning approach, 2019. https://doi.org/10.48550/arXiv.1901.01774.

[7] W.K.O. Ho, B.-S. Tang, S.W. Wong. Predicting property prices with machine learning algorithms. *J. Prop. Res.*, **38**(1):48–70, 2020. https://doi.org/10.1080/09599916.2020.1832558.

[8] S. Lu, Z. Li, Z. Qin, X. Yang, R.S.M. Goh. A hybrid regression technique for house prices prediction. In *International Conference on Industrial Engineering and Engineering Management*, IEEM, 2017.

[9] J. Oxenstierna. Predicting house prices using ensemble learning with cluster aggregations. In *Proceedings of the 2019 11th International Conference on Machine Learning and Computing*, ICMLC, 2017.

[10] P.-F. Pai, W.-C. Wang. Using machine learning models and actual transaction data for predicting real estate prices. *Appl. Sci.*, **10**(17):5832, 2020. https://doi.org/10.3390/app10175832.

[11] B. Park, J.K. Bae. Using machine learning algorithms for housing price prediction: The case of fairfax county, virginia housing data. *Exp. Syst. Appl.*, pp. 2928–2934, 2015. https://doi.org/10.1016/j.eswa.2014.11.040.

[12] T.D. Phan. Housing price prediction using machine learning algorithms: The case of melbourne city, australia. In *International Conference on Industrial Engineering and Engineering Management*, IEEM, 2018. https://doi.org/10.1109/iCMLDE.2018.00017.

[13] S.J. Semnani, H. Rezaei. House price prediction using satellite imagery. *Intel. Syst. Appl.*, **14**, 2021. https://doi.org/10.1016/j.iswa.2022.200081.

[14] Q. Truong, M. Nguyen, H. Dang, B. Mei. Housing price prediction via improved machine learning techniques. *Procedia Comput. Sci.*, **174**:433–442, 2020. https://doi.org/10.1016/j.procs.2020.06.111.

SUMMARY

Real estate valuation using machine learning techniques

S. Adomavičius

In this work, the methods of artificial intelligence are analyzed in order to perform a more accurate value prediction of the apartments sold in Vilnius city and district. The work uses publicly available information about apartments put for sale on Aruodas.lt, which is collected in an automated way. The information that is collected consists of a text – description of the apartment for sale, photos – photos placed in the ad, and general information provided in the ad – price, location, apartment size, various features of the apartment and more. A constant problem in real estate valuation is the overvaluation of low-value objects and/or underestimation of high-value objects. When dealing with regression problems, we often have data on most average objects, but never enough of cheap

and luxurious ones. For this reason, estimating the value of most properties is easier than cheap or expensive. However, as the methods of artificial intelligence evolve and information becomes more and more available, our ability to better value this type of housing increases. It is hoped that the information contained in the photos and text will allow for a better forecast of the value of housing by better valuing both cheap and expensive apartments. In the first part of the work, a review of the literature is performed where other authors have examined the possibilities of using artificial intelligence to predict the value of housing. The second part describes the research methods that will be applied in the work and presents the information gathering strategy. In the third part, a study is conducted, during which feature engineering is performed first, followed by model training and optimization. Finally, the results of the best model with 13.74 MAPE and 33,307 RMSE are presented along with the conclusions of the work.

Keywords: machine learning; regression; housing price prediction; text analysis; vision analysis; clustering; aruodas.lt