

ՀՀ ԳԱԱ ԻՆՖՈՐՄԱՏԻԿԱՅԻ ԵՎ ԱՎՏՈՄԱՏԱՑՄԱՆ ՊՐՈԲԼԵՄՆԵՐԻ ԻՆՏԻՏՈՒՏ

Օղլուկյան Արթուր Կարապետի

ՏԵՂԱԲԱՇԽՎԱԾ ԵՎ ՉՈՒԳԱՐԵՌ ՄԵՔԵՆԱՅԱԿԱՆ ՈՒՍՈՒՑՄԱՆ ԼՈՒԾՈՒՄՆԵՐԻ
ՆԱԽԱԳԾՈՒՄ

Ե.13.04 - «Հաշվողական մեքենաների, համալիրների, համակարգերի և ցանցերի
մաթեմատիկական և ծրագրային ապահովում» մասնագիտությամբ տեխնիկական
գիտությունների թեկնածուի գիտական աստիճանի հայցման ատենախոսության
սեղմագիր

Երևան 2025

INSTITUTE FOR INFORMATICS AND AUTOMATION PROBLEMS OF THE NAS RA

Oghlukyan Artur Karapet

DESIGNING DISTRIBUTED AND PARALLEL MACHINE LEARNING SOLUTIONS

An abstract of dissertation for obtaining a Ph.D. degree in Technical Sciences on specialty
13.04 "Provision of mathematical equations and programs for computing machines,
complexes, networks and systems"

Yerevan 2025

Ատենախոսության թեման հաստատվել է ՀՀ ԳԱԱ Ինֆորմատիկայի և
Ավտոմատացման Պրոբլեմների Ինստիտուտում

Գիտական ղեկավար՝ ֆիզ. մաթ. գիտ. դոկտոր Լ. Ասլանյան

Պաշտոնական ընդդիմախոսներ՝ _____

Առաջատար կազմակերպություն՝ _____

Պաշտպանությունը կայանալու է _____ մասնագիտական խորհրդում

Ատենախոսությանը կարելի է ծանոթանալ ՀՀ ԳԱԱ ԻԱՊԻ գրադարանում

Սեղմագիրն առաքված է _____ ամսաթիվ

Մասնագիտական խորհրդի գիտական քարտուղար _____

The topic of the dissertation was approved at the Institute for Informatics and Automation
Problems of NAS RA

Scientific Supervisor: L. Aslanyan, D. Phys. Math. S.

Official Opponents: _____

Leading Organization: _____

The Defense will take place _____

The Dissertation is available at the library of IIAP NAS RA.

The abstract was delivered on _____

Scientific Secretary of the Specialized Council _____

Ատենախոսության ընդհանուր բնութագիր

Թեմայի արդիականությունը

Արհեստական Բանականության և մեծ տվյալների դարաշրջանում մեքենայական ուսուցման ալգորիթմների արդյունավետ կիրառությունը տարբեր բաշխումների ենթակա տվյալների վրա հանդիսանում է կենսական խնդիր: Հատկապես սա կարևոր է PAC (Probably Approximately Correct) ուսուցման տեսական մոդելի շրջանակներում, որտեղ ալգորիթմները պետք է ապահովեն կարգավորելի ճշգրտություն և վստահություն կամայական բաշխումների պարագայում: PAC մոդելը, որը ներմուծվել է Վալիանտի կողմից [1], ֆորմալացնում է մեքենայական ուսուցման հիմնական գաղափարները և սահմանում է այն պայմանները, որոնց դեպքում ալգորիթմը կարող է արդյունավետ սովորել (ուժեղ դասակարգիչ):

Boosting ալգորիթմները, մասնավորապես AdaBoost-ը, հանդիսանում են PAC ուսուցման խնդիրների մասնակի լուծում՝ թույլ տալով համակցել թույլ դասակարգիչների հավաքածուները և ստեղծել ուժեղ դասակարգիչ տեսականորեն երաշխավորված կատարողականությամբ: Շափիրեի և Ֆրեյնդի տեսական արդյունքները ցույց են տվել, որ եթե գոյություն ունի թույլ սովորող ալգորիթմներ, որոնց արդյունավետությունը գերազանցում է պատահական գուշակումը, ապա boosting-ը կարող է կառուցել բարձր ճշգրտությամբ ուժեղ դասակարգիչ:

Սակայն առկա boosting մեթոդների հիմնական սահմանափակումը դրանց հաջորդական բնույթն է՝ յուրաքանչյուր թույլ մոդելը պետք է սովորեցվի նախորդների հիման վրա, ինչը դժվարացնում է մեծածավալ տվյալների արդյունավետ մշակումը: Այս սահմանափակումը հատկապես կրիտիկական է դառնում ժամանակակից կիրառություններում, որտեղ տվյալների ծավալները հասնում են պետաբայթների, ինչը պահանջում է ժամանակատար մշակում: Բացի այդ, ավանդական AdaBoost-ի հաղորդակցական բարդությունը և համաժամեցման պահանջները բախվածության պարագայում դարձնում են այն ոչ պրակտիկ մեծ համակարգերի համար, որտեղ ցանցային ուշացումները և հանգույցների անկանխատեսելի վարքագիծը կարող են խիստ ազդել ալգորիթմի զուգամիտման վրա:

Աշխատանքի նպատակն ու խնդիրները

Հիմնական նպատակ: AdaBoost դասի ալգորիթմների արդյունավետ բաշխված կառուցվածքով իրականացման մշակումն է՝ օգտագործելով բաշխվածության և զուգահեռացման ծրագրային տեխնոլոգիաներ, օրինակ MPI (Message Passing Interface) տեխնոլոգիան և ադապտիվ հաղորդակցական մեխանիզմներ:

Խնդիրները:

1. Բաշխված AdaBoost ալգորիթմների տեսական վերլուծություն
2. MPI-հիմնված ճարտարապետության ներդրում
3. Ադապտիվ հաղորդակցական ժամանակացույցի ներդրում
4. Բուժերային մեխանիզմների իրականացում տեղական մոդելների կառավարման համար
5. Փորձարարական գնահատման իրականացում

Յետազոտման օբյեկտը

Յետազոտության օբյեկտն է AdaBoost boosting ալգորիթմի բաշխված իրականացումը՝ կենտրոնաւաւով հաղորդակցական արդյունավետության և մասշտաբավորելիության բարելավման վրա: Յետազոտությունը ուսումնասիրում է ավանդական AdaBoost ալգորիթմի առանձնահատկությունները և սահմանափակումները բաշխված միջավայրերում, որտեղ տվյալները բաժանված են մի քանի հաշվողական հանգույցների միջև:

Մասնավորապես, օբյեկտը ներառում է AdaBoost-ի հաջորդական բնույթի վերափոխումը բաշխված մոդելի, որտեղ յուրաքանչյուր հանգույց կարող է անկախ մշակել տվյալների իր հատվածը՝ պարբերաբար համաժամեցվելով մյուս հանգույցների հետ: Յետազոտության կենտրոնում է նաև թույլ դասակարգիչների տեղական ուսուցման և գլոբալ ինտեգրման մեխանիզմների օպտիմալացումը, ինչպես նաև հաղորդակցական ծախսերի նվազեցումը՝ պահպանելով տեսական երաշխիքները:

Յետազոտության շրջանակներում ուսումնասիրվում են ադապտիվ համաժամեցման մեթոդները, բուժերային մեխանիզմները և MPI-հիմնված բաշխված ճարտարապետությունները՝ նպատակ ունենալով ստեղծել մասշտաբավորվող լուծում, որը կարող է արդյունավետ աշխատել տարբեր չափերի կլաստերներում և հարմարվել տարբեր ցանցային պայմանների:

Յետազոտման մեթոդները

- Մեքենայական ուսուցման տեսական մեթոդներ
- Բաշխված համակարգերի նախագծում
- MPI (Message Passing Interface) ծրագրավորում
- Գործնական փորձարկում supercomputing միջավայրերում
- Համեմատական վերլուծություն

Արդյունքների գիտական նորույթը

1. **Ադապտիվ համաժամեցման մեխանիզմի մշակում.** Նորարարական մոտեցում համաժամեցման միջակայքի դինամիկ կարգավորման համար՝ հիմնվելով սխալի փոփոխման վրա
2. **Բուժերային ռազմավարության ներդրում.** լոկալ/տեղական մոդելների արդյունավետ կուտակում և կառավարում
3. **Սխալի չափի վրա հիմնված ադապտիվ սինխրոնացում.** սխալի հարաբերակցության հիման վրա հաղորդակցության օպտիմալացում

Ստացված արդյունքների կիրառական նշանակությունը

Առաջարկվող բաշխված AdaBoost ալգորիթը ցույց է տվել նշանակալի բարելավումներ՝ ապահովելով 35% հաղորդակցային ծախսերի նվազում մշակված տվյալների վրա և 32% ուսուցման ժամանակի կրճատում՝ պահպանելով ճշգրտությունը: Ալգորիթը ցուցաբերում է գծային մասշտաբավորելիություն մինչև 32 հանգույց և կարող է լայնորեն կիրառվել մեծածավալ տվյալների դասակարգման խնդիրներում, ինչպիսիք են ֆինանսական ռիսկերի գնահատումը, բժշկական տվյալների վերլուծությունը, օնլայն գովազդների օպտիմալացումը և առաջարկությունների համակարգերը:

Ներդրումներ

Ատենախոսությունն անդրադառնում է բաշխված մեքենայական ուսուցման արդյունավետության բարձրացմանը՝ Adaboost ալգորիթի հիման վրա ադապտիվ հաղորդակցական պլանավորման մեխանիզմների կիրառման միջոցով: Նորարարական ներդրումը կայանում է բաշխված Adaboost-ի մոդիֆիկացված տարբերակի առաջարկման մեջ, որը նախատեսված է ադապտիվ հաղորդակցության հաճախականության վերահսկման համար՝ հիմնված մոդելային սխալի դինամիկայի վրա: Առաջարկվում են տեսական վերլուծություններ և փորձարարական արդյունքներ, որոնք ապացուցում են Նոր մոտեցման արդյունավետությունը՝ նվազեցնելով հաղորդակցման ծախսերը մինչև 30%՝ առանց դասակարգման ճշգրտության կորուստների:

Պաշտպանությանը ներկայացվում են հետևյալ դրույթները

1. MPI մեխանիզմի վրա հիմնված բաշխված AdaBoost ճարտարապետություն՝ ադապտիվ հաղորդակցական մեխանիզմներով
2. Բուժերային ռազմավարություն՝ տեղական մոդելների թարմացումների արդյունավետ կառավարման համար
3. Սխալի չափի վրա հիմնված ադապտիվ սինխրոնացման ալգորիթմ՝ հաղորդակցության նվազագույնացման համար
4. Փորձարարական գնահատման արդյունքներ՝ ցուցադրող գծային մասշտաբավորելիություն և արդյունավետություն

Ստացված արդյունքների ապրոբացիան

Արդյունքները փորձարկվել են տարբեր առկա և սինթետիկ տվյալների բազմությունների վրա: Ալգորիթմի արդյունավետությունը բարձրացվել է supercomputing միջավայրի կիրառմամբ:

Հրատարակությունները

1. A. Oghlukyan, "Adaptive communication for scalable distributed AdaBoost framework," Pattern Recognition and Image Analysis, vol. 35, no. 2, pp. 119–128, 2025.
2. V. Mkrtchyan and A. Oghlukyan, "Prospects for further improvement of machine learning systems," ALTERNATIVE quarterly academic journal, 2023, no. 2 pp. 40-45.
3. A. Oghlukyan and G. Karapetyan, "Information technologies and developing a single page web application using the React library," ALTERNATIVE quarterly academic journal, 2022, no 1, pp. 34-39.
4. M. Yeghiazaryan and A. Oghlukyan, "On some issues related to machine learning in 2022," ALTERNATIVE quarterly academic journal, 2022, no. 2, pp. 98-103.
5. A. Martirosyan and A. Oghlukyan, "Economic impact of developing of web technologies", ALTERNATIVE quarterly academic journal, 2022, no. 1, pp. 26-33

Աշխատանքները թեզի շրջանակներում շարունակվում են և երեք լրացուցիչ հոդվածների տեսքով ներկայացված են ատենախոսությունից դուրս և թեմայի շրջանակներում ստացված նոր փորձարարական արդյունքները.

1. A. Ogulkyan, Nuria Gómez Blas, “Integrating Asynchronous AdaBoost into Federated Learning: Five Real- World Applications”, Կոմպյուտերային գիտության մաթեմատիկական խնդիրներ
2. A. Ogulkyan, Luis Fernando de Mingo López, “Infrequent Synchronization in Distributed AdaBoost”, Կոմպյուտերային գիտության մաթեմատիկական խնդիրներ
3. L. Aslanyan, A. Ogulkyan, “Analysis of Distributed Boosting”, Կոմպյուտերային գիտության մաթեմատիկական խնդիրներ

Աշխատանքի կառուցվածքը և ծավալը

Ատենախոսությունը բաղկացած է ներածությունից, չորս գլխից, եզրակացությունից և գրականության ցանկից: Ընդհանուր ծավալը կազմում է [X] էջ, ներառյալ [Y] նկար, [Z] աղյուսակ և [W] գրականական աղբյուր:

Գլուխների բովանդակությունը:

- Գլուխ 1: Հիմնական հասկացություններ և տիրույթի վերլուծություն
- Գլուխ 2: Տեղաբաշխված համակարգեր
- Գլուխ 3: Բաշխված AdaBoost ալգորիթմներ և դրանց փորձարարական գնահատումը
- Գլուխ 4: Առաջարկվող բաշխված AdaBoost համակարգ

Աշխատանքի բովանդակությունը

Գլուխ 1. Հիմնական հասկացություններ և տիրույթի վերլուծություն

Ժամանակակից մեքենայական ուսուցման ոլորտում **Boosting** մեթոդները առանձնանում են իրենց բարձր արդյունավետությամբ, հատկապես՝ թույլ դասակարգիչների հիման վրա ուժեղ մոդելների կառուցման կարողությամբ: Նրանց հաջող կիրառումը բազմաթիվ կիրառական խնդիրներում հանգեցրել է խորքային ուսումնասիրությունների ինչպես տեսական, այնպես էլ գործնական մակարդակներում: Այս գլխում ներկայացվում են Boosting ալգորիթմների հիմնական գաղափարները, համեմատական վերլուծությունը առավել հայտնի տարբերակների միջև՝ առանձնահատուկ ուշադրությամբ՝ **AdaBoost**, **Gradient Boosting** և **XGBoost** մեթոդների նկատմամբ, ինչպես նաև քննարկվում են այդ մեթոդների կիրառության սահմանափակումները՝ մեծածավալ տվյալների (Big Data) մշակման համատեքստում:

1.1 Boosting ալգորիթմների տեսական հիմունքներ

Boosting-ը հանդիսանում է համախմբող ուսուցման (ensemble learning) ենթաբաժիններից մեկը, որի հիմնական նպատակն է վերածել **թույլ սովորողներ** (weak learners)՝ բարձր սխալով, **ուժեղ սովորողների** (strong learners), որոնք ունեն բարձր ճշգրտություն: Թույլ սովորողը, սահմանմամբ, այն մոդելն է, որի կատարողականը փոքր-ինչ գերազանցում է պատահական գուշակումը, մինչդեռ ուժեղ սովորողը զգալիորեն գերազանցում է այդ շեմը:

AdaBoost (Adaptive Boosting) ալգորիթմը, որը ներկայացվել է Յոավ Ֆրենկի և Ռոբերտ Շափիրոյի կողմից 1996 թվականին, հիմնված է այն գաղափարի վրա, որ հաջորդական դասակարգիչները հետևողականորեն շեշտադրում են նախորդների սխալների վրա՝ յուրաքանչյուր նմուշին վերագրելով քաշ: Յուրաքանչյուր հաջորդ դասակարգիչ առավել շատ կենտրոնանում է այն տվյալների վրա, որոնք սխալ էին դասակարգվել նախորդ փուլում: Վերջում բոլոր դասակարգիչների արդյունքները միավորվում են քաշված մեխանիզմով՝ ստանալով վերջնական արդյունք:

Ի տարբերություն AdaBoost-ի, **Gradient Boosting** մեթոդը (Friedman, 2001) հիմնված է կորուստի ֆունկցիայի գրադիենտով օպտիմալացման գաղափարի վրա: Այն յուրաքանչյուր փուլում փորձում է նվազեցնել ընդհանուր սխալը՝ մոտավորելով կորուստի գրադիենտը որոշակի թույլ մոդելով, սովորաբար՝ որոշման ծառով: Այս մոտեցումը դարձնում է մեթոդը առավել ճկուն, քանի որ կարելի է ընտրել տարբեր կորուստի ֆունկցիաներ՝ կախված խնդրից:

XGBoost-ը (eXtreme Gradient Boosting) համարվում է Gradient Boosting-ի բարձր արդյունավետությամբ իրականացում, որն ապահովում է մոդելավորման արագություն և ճշգրտություն՝ շնորհիվ համակարգչային օպտիմալացման հնարքների, ինչպես են՝ ծառերի մակարդակով պարբերականացման մեխանիզմներ, կանխասահմանված քաշման սանդղակներ, և բազմամիջամտական տվյալների կառավարում:

1.2 Boosting-ի կիրառություններ ու սահմանափակումներ մեծ տվյալների պայմաններում

Boosting ալգորիթմները լայն կիրառություն ունեն տարբեր ոլորտներում՝ սկսած ֆինանսական մոդելավորումից և բժշկական ախտորոշումից մինչև բնական լեզվի մշակում և recommendation համակարգեր: Հատկապես **XGBoost**-ը աչքի է ընկել բազմաթիվ մրցույթներում՝ Kaggle-ի հայտնի դեպքերում՝ հանդես գալով որպես հաղթող մոդելների հիմնական բաղադրիչ:

Այնուամենայնիվ, Boosting մեթոդների կիրառությունը **մեծ տվյալների** պարագայում բախվում է մի շարք խնդիրների: Նախ, դասակարգիչների հաջորդական ուսուցման բնույթը սահմանափակում է համակարգի զուգահեռացման (parallelization) հնարավորությունները: Բացի այդ, հիշողության սպառումը և հաղորդակցական ծախսերը՝ տվյալները փոխանցելու և մոդելները համադրելու ընթացքում, էապես աճում են տվյալների ծավալի մեծացման հետ:

AdaBoost-ի դեպքում այդ սահմանափակումներն առավել ակնհայտ են, քանի որ մեթոդը կախված է նմուշների քաշերի մշտական թարմացումից և խիստ հաջորդական ինտերակցիաներից: Իսկ Gradient Boosting և XGBoost մեթոդները, չնայած զուգահեռացման որոշակի մեխանիզմներին, նույնպես դառնում են հաշվարկային առումով ծանրաբեռն, երբ գործ ունենք բաշխված միջավայրի հետ:

Այսպիսով, Boosting մեթոդների կիրառելիությունը մեծ տվյալների միջավայրում պահանջում է հատուկ մշակված ճարտարապետություններ, որոնք համատեղում են արդյունավետ ուսուցում՝ նվազագույն հաղորդակցական ծախսերով: Այդ նպատակով առաջարկվում է ուսումնասիրել բաշխված ուսուցման ճարտարապետությունները, ինչպես նաև զարգացնել **ադապտիվ հաղորդակցման պլանավորման** մեխանիզմներ, որոնց միջոցով հնարավոր կլինի զուգահեռաբար հաշվարկել և ուսուցանել, միաժամանակ նվազեցնելով համաժամեցման անհրաժեշտությունը:

Գլուխ 2. Տեղաբաշխված համակարգեր

Տեղաբաշխված համակարգերը դարձել են ժամանակակից տվյալագիտության և մեքենայական ուսուցման անբաժանելի մասը: Տվյալների ծավալների աննախադեպ աճը պահանջում է նոր լուծումներ, որոնք կարող են ապահովել միաժամանակ բարձր հաշվարկային հզորություն, արդյունավետ հիշողության կառավարում և ցածր հաղորդակցական ծանրաբեռնվածություն: Այս գլուխում ներկայացվում են բաշխված հաշվարկի հիմնարար մոդելները, դրանց կապակցված ցանցային տոպոլոգիաները, հաղորդակցման և համաժամեցման հիմնական մարտահրավերները, ինչպես նաև գործնականում կիրառվող ծրագրային հարթակները:

2.1 Գանցային տոպոլոգիաներ և բաշխված հաշվարկի հիմնարար մոդելներ

Տեղաբաշխված ուսուցման գործընթացում տվյալները և մոդելները բաժանվում են բազմաթիվ հանգույցների (nodes) միջև, որոնք փոխկապակցված են տարբեր ցանցային տոպոլոգիաներով: Այս համակարգերը կարող են կազմված լինել հզոր սերվերային հանգույցներից կամ՝ ավելի թեթև հաշվարկային միավորներից՝ միացված ինտերնետի միջոցով: Տարբեր տոպոլոգիաներ՝ ինչպես օրինակ աստղաձև (star), շրջայական (ring), ցանցային (mesh) կամ ծառաձև (tree), առաջարկում են տարբեր փոխգիշումներ թողունակության, ուշացման և աղետավտանգության տեսանկյունից:

Տեղաբաշխված հաշվարկի ամենատարածված մոդելներից են.

- **MapReduce** մոդելը՝ ներդրված Google-ի կողմից, որը հիմնված է տվյալների տարանջատված մշակման վրա: Այն բաղկացած է երկու փուլից՝ **Map** (քարտեզագրում) և **Reduce** (նվազեցում), որոնք հնարավոր են գործարկել զուգահեռաբար՝ առանց հանգույցների միջև ինտենսիվ հաղորդակցման: Մոդելը պարզ է, սակայն սահմանափակ է բարդ կախվածություններով խնդիրների դեպքում:
- **Parameter Server (PS)** մոդելը՝ դասական մոտեցում է բաշխված ուսուցման համար, որտեղ առկա են երկու հիմնական բաղադրիչ՝ **աշխատողներ (workers)** և **պարամետրերի սպասարկիչներ (servers)**: Աշխատողները հաշվարկում են մոդելի թարմացումները, իսկ սպասարկիչները պահպանում և համաժամեցնում են ընդհանուր պարամետրերը: Այս մոդելը առավելապես կիրառվում է մոդելային պարամետրերի մեծ քանակի պարագայում, սակայն ենթակա է հաղորդակցական ծանրաբեռնվածության խնդիրների:

- **All-Reduce** մոդելը՝ թույլ է տալիս բոլոր հանգույցներին միաժամանակ հաղորդակցել միմյանց հետ՝ իրենց տեղական արդյունքները համադրելու և հանրագումարային արդյունք ստանալու նպատակով: Այն իդեալական է համաժամեցված մոդելների ուսուցման համար և լայնորեն կիրառվում է թվային խորքային ուսուցման գրադարաններում (օր.՝ Horovod):

Այս մոդելներն ունեն տարբեր հաշվարկային ու ցանցային բնութագրեր՝ յուրաքանչյուրը ավելի հարմար տարբեր կիրառական սցենարների համար:

2.2 Հաղորդակցում և համաժամեցում, սինխրոն/ասինխրոն, թողունակության ու ուշացման խնդիրներ

Տեղաբաշխված համակարգերում տվյալների փոխանակումը հիմնականում տեղի է ունենում ցանցային հաղորդակցության միջոցով: Հաշվի առնելով հանգույցների տեղաբաշխված բնույթը՝ հաղորդակցման արդյունավետությունը դառնում է ուսուցման ընդհանուր արագության որոշիչ գործոն: Գոյություն ունեն երկու հիմնական համաժամեցման մեխանիզմ՝ **սինխրոն (synchronous)** և **ասինխրոն (asynchronous)** ուսուցում:

Սինխրոն ուսուցման ժամանակ բոլոր հանգույցները սպասում են միմյանց արդյունքներին՝ նախքան հաջորդ քայլին անցնելը: Սա ապահովում է բարձր ճշգրտություն և կանխատեսելի կոնվերգենցիա, սակայն խիստ կախված է ցանցի թողունակությունից և հանգույցների հզորությունից: Ավելին, մեկ դանդաղ հանգույց (straggler) կարող է դանդաղեցնել ամբողջ համակարգը:

Ասինխրոն ուսուցման ժամանակ յուրաքանչյուր հանգույց աշխատում է անկախ՝ ուղարկելով և ստանալով պարամետրերի թարմացումներ առանց մյուսներին սպասելու: Այս մոտեցումը ավելի արդյունավետ է ցածր թողունակությամբ ցանցերում, սակայն կարող է հանգեցնել մոդելի անկայուն ուսուցման՝ պարամետրերի նավագաժամանակյա համաժամեցման պատճառով:

Հաղորդակցման գործընթացի երկու հիմնական խնդիրներն են՝

- **Թողունակություն (bandwidth)** – սահմանափակում, թե որքան տվյալ կարող է փոխանցվել միավոր ժամանակում:
- **Ուշացում (latency)** – ժամանակը, որը պահանջվում է տվյալների մեկ փաթեթի փոխանցման համար:

Մոդելների ուսուցման ընթացքում, հատկապես պարամետրերի հաճախակի փոխանակման դեպքում, այս երկու գործոնները կարող են զգալիորեն դանդաղեցնել

ուսուցման գործընթացը: Արդյունքում, արմատանում է **ադապտիվ հաղորդակցման** մեխանիզմների անհրաժեշտությունը՝ որոնցում հանգույցները կարող են կարգավորել հաղորդակցության ինտենսիվությունը՝ ըստ ուսուցման փուլում հասած վիճակի:

2.3 Գործնական հարթակներ (MPI, Horovod, Spark MLlib, XGBoost-Distributed)

Տեղաբաշխված հաշվարկը կիրառական մակարդակում իրականացվում է հատուկ մշակված հարթակների միջոցով: Ստորև ներկայացված են առավել տարածված և կիրառելի գործիքակազմերը՝ համապատասխան ուսուցման մոդելներին:

- **MPI (Message Passing Interface)**՝ դասական կոմունիկացիոն ինտերֆեյս է բաշխված համակարգերի համար: Այն առաջարկում է ցածր մակարդակի API տվյալների փոխանցման համար՝ միաժամանակ ապահովելով սինխրոն և ասինխրոն հաղորդակցության հնարավորություններ: MPI-ն առավել հարմար է հստակ կառավարվող համակարգերում, ինչպիսին HPC (High Performance Computing) կլաստերներն են:
- **Horovod**՝ զարգացած Uber ընկերության կողմից, դա բարձր մակարդակի գրադարան է, որը կառուցված է MPI-ի և NCCL-ի (NVIDIA Collective Communications Library) վրա: Այն ապահովում է **All-Reduce** վրա հիմնված արդյունավետ մոդելային համաժամեցում, հատկապես՝ TensorFlow և PyTorch շրջանակներում: Horovod-ը թույլ է տալիս արագորեն տարածել մեծածավալ մոդելների ուսուցումը՝ բազմապրոցեսոր միջավայրում:
- **Spark MLlib**՝ Apache Spark-ի մաս է կազմում՝ տրամադրելով բաշխված ուսուցման ալգորիթմների փաթեթ՝ հիմնված MapReduce սկզբունքի վրա: Չնայած այն չի համարվում բարձր արդյունավետության լուծում խորը ուսուցման համար, սակայն հարմար է դասական մեքենայական ուսուցման ալգորիթմների համար՝ հաշվի առնելով տվյալների նախավերաբերման (preprocessing) ինտեգրացիան:
- **XGBoost-Distributed**՝ XGBoost գրադարանի բաշխված տարբերակն է, որը կառուցված է Rabbit հաղորդակցման շերտի վրա: Այն թույլ է տալիս դասակարգման և ռեգրեսիայի առաջադրանքներում օգտագործել բազմաթիվ հանգույցներ՝ միաժամանակ պահպանելով մոդելի ճշգրտությունը: XGBoost-ը ներառում է խելացի տվյալների բաշխման և պարամետրերի համաժամեցման մեխանիզմներ՝ նվազեցնելով թողունակության վրա բեռը:

Այս հարթակներն առաջարկում են տարբեր մակարդակների վերահսկում կախված կիրառության տիպից և հաշվարկային ռեսուրսների հասանելիությունից:

Արդյունավետ տեղաբաշխված ուսուցման հասնելու համար անհրաժեշտ է հաշվի առնել ինչպես տվյալների կառուցվածքը, այնպես էլ հաղորդակցության և համակարգչային ճարտարապետության գործոնները:

Գլուխ 3. Բաշխված AdaBoost ալգորիթներ և դրանց փորձարարական գնահատումը

Boosting-ի դասի ալգորիթները, և հատկապես՝ AdaBoost-ը, համարվում են դասակարգման դասական մոտեցումներից մեկը՝ իրենց պարզ կառուցվածքի և բարձր արդյունավետության շնորհիվ: Դրանց հաջողությունը պայմանավորված է թույլ մոդելների հաջորդական համադրմամբ, որի արդյունքում կառուցվում է մեկ միասնական ուժեղ դասակարգիչ: Սակայն ժամանակակից խոշորածավալ տվյալների և բարձր չափականությամբ տարածքներում ալգորիթի կիրառումը բախվում է մի շարք սահմանափակումների, ինչպիսիք են՝ մեծ հաշվարկային բեռը, հիշողության ծավալը և մոդելի սանդղավումայնության (scalability) խնդիրները: Այդ իսկ պատճառով վերջին տարիներին կարևորություն է ստացել AdaBoost-ի բաշխված իրագործումը:

AdaBoost-ի տեսական հիմքը ենթադրում է հետևյալ մեխանիզմը. նախ, բոլոր ուսուցման նմուշները ստանում են սկզբնական հավասար քաշեր: Այնուհետև, յուրաքանչյուր հաջորդ փուլում թարմացվում են այդ քաշերը՝ ելնելով մոդելի կատարած սխալից. սխալ դասակարգված նմուշների քաշերն աճում են, իսկ ճիշտ դասակարգվածներինը՝ նվազում: Վերջնական արդյունքը հանդիսանում է բոլոր փուլերում ստացված թույլ մոդելների քաշված համադրությունը: Այս մեխանիզմը բերում է սխալի աստիճանական նվազեցման և տեսականորեն ապահովում է գեներալիզացիայի լավ ունակություններ: Սակայն ալգորիթի ներսում գոյություն ունեցող հաջորդական կապերը դժվարացնում են նրա զուգահեռ կամ բաշխված իրագործումը:

Բաշխվածության հիմնական մարտահրավերն այն է, որ յուրաքանչյուր փուլում մոդելի կառուցումը և նմուշների քաշերի թարմացումը կախված են նախորդ փուլից ստացված արդյունքներից: Այս շարքային կախվածությունը պահանջում է համակարգում ապահովել տվյալների մշտական համաժամեցում: Հիմնական ռազմավարություններից մեկը տվյալների բաշխումն է՝ ըստ հանգույցների, որտեղ յուրաքանչյուր հանգույցում իրականացվում է մասնակի ուսուցում: Այս դեպքում հարկավոր է պարբերաբար համադրել բոլոր հանգույցների արդյունքները և միասին թարմացնել քաշերը: Հաճախ կիրառվում է համաժամեցման մեթոդ՝ օգտագործելով կամ կենտրոնացված սերվեր, կամ All-Reduce մեխանիզմ, ինչը սակայն մեծացնում է հաղորդակցման ծախսը:

Որպես այլընտրանք, որոշ համակարգեր առաջարկում են թույլ մոդելների զուգահեռ կառուցում՝ մոդելամետ պարալելիզմի սկզբունքով: Այստեղ հնարավոր է տարբեր

հանգույցներում ուսուցանել տարբեր թույլ դասակարգիչներ և ապա դրանք միավորել՝ ստեղծելով մեկ միասնական մոդել: Չնայած այս մոտեցումը տեսականորեն թույլ է տալիս բարելավել ուսուցման արագությունը, այն խիստ կախված է թույլ մոդելների կառուցման բնույթից և դրանց միջև կախվածությունների բացակայությունից:

Վերջին շրջանում դիտվում են փորձեր նաև ասինխրոն ուսուցման ուղղությամբ: Այդպիսի համակարգերում որոշ հանգույցներ կարող են ուսուցանել առանց սպասելու ընդհանուր արդյունքներին՝ կիրառելով միջինացնող կամ պահեստային (stale) մեխանիզմներ: Թեև այս մեթոդները որոշ չափով նվազեցնում են հաղորդակցման բեռը, դրանք կարող են վնասել մոդելի ընդհանուր ճշգրտությանը, հատկապես՝ զգայուն տվյալների պարագայում:

Գործնականում բաշխված AdaBoost-ի իրագործմանն ուղղված մի քանի համակարգեր արդեն գոյություն ունեն: Դրանցից են՝ MPI-ի հիման վրա կառուցված DistBoost համակարգը, որն ապահովում է բաշերի կենտրոնացված թարմացում և սինխրոն ուսուցում, ինչպես նաև Gradient Boosting-ի վրա հիմնված XGBoost-ը և LightGBM-ը, որոնք իրագործում են Boosting-ի այլ տարբերակներ՝ ապահովելով բարձր արդյունավետություն մեծ տվյալների պարագայում: Թեև վերջինները չունեն AdaBoost-ի ամբողջական մեխանիզմը, դրանց ճարտարապետությունը արժեքավոր է բաշխված ուսուցման տեսանկյունից: Մասնավորապես, XGBoost-ը կիրառում է Rabbit հաղորդակցման շերտը, որը թույլ է տալիս արդյունավետ All-Reduce գործողություններ և ապահովում է հուսալիություն՝ հանգույցների հնարավոր խափանումների պայմաններում:

Այս համատեքստում զարգանում են նաև հաղորդակցման հարմարեցվող սխեմաներ, որոնց նպատակը հաղորդակցման հաճախականության և հաշվարկային ճշգրտության միջև հավասարակշռություն գտնելն է: Օրինակ, որոշ մեթոդներ ուսուցման սկզբնական փուլերում կիրառում են հաճախակի համաժամեցում, իսկ հետագայում՝ ավելի նոսր, երբ մոդելը կայունանում է: Հետաքրքիր զարգացում է նաև բաշերի քվանտացման և սեղմման մեթոդների կիրառումը, որը նպաստում է ցանցային հաղորդման ծավալի կրճատմանը՝ առանց զգալիորեն վնասելու ճշգրտությանը:

Չնայած այս առաջընթացին՝ մի շարք բաց հարցեր մնում են լուծման փուլում: Առաջին հերթին՝ ուսուցման սխալի վերահսկումը բաշխված պայմաններում բարդ է իրականացնել, հատկապես երբ կիրառվում են ասինխրոն մեխանիզմներ: Բացի այդ, հանգույցների բեռնվածության հավասարակշռումը մնում է տեխնիկական լուրջ մարտահրավեր: Հատկապես դժվար է ապահովել, որ բոլոր հանգույցներն աշխատեն նույն տեմպով և ոչ մեկը չդառնա համակարգի արագությունը սահմանափակող հանգույց:

Գոյություն ունեն նաև հետազոտական հարցեր մետաուսուցման կիրառման հնարավորությունների վերաբերյալ: Այդ մոտեցումները կարող են օգնել ընտրել

առավել հարմար թույլ դասակարգիչներ յուրաքանչյուր փուլում կամ նախապես գնահատել, թե որ նմուշներն են առավել արժեքավոր հաջորդ փուլում: Մեկ այլ հեռանկարային ուղղություն է բաշխված AdaBoost-ի տեղաբաշխված իրագործումը՝ սահմանափակ ռեսուրսներով սարքերում (օրինակ՝ սենսորային ցանցեր կամ շարժական սարքեր), որտեղ անհրաժեշտ է կոմպակտ մոդելներ և օպտիմալ հաղորդակցություն:

Ընդհանրացնելով՝ կարելի է ասել, որ AdaBoost-ի բաշխված իրագործումը, թեև բարդ է տեսական և ճարտարապետական առումներով, ունի զգալի ներուժ ինչպես գիտական, այնպես էլ կիրառական ոլորտներում: Ազդրիթմի աստիճանական ճշգրտման բնույթը լավ համադրվում է հարմարվող բաշխման մեխանիզմների հետ, իսկ նորագույն հաղորդակցման սխեմաները թույլ են տալիս մեղմել շարքայնության ներմուծած սահմանափակումները: Հետագա առաջընթացը կախված կլինի թե՛ ազդրիթմների օպտիմալացմամբ, թե՛ ծրագրային ճարտարապետությունների կատարելագործմամբ:

Գլուխ 4. Առաջարկվող բաշխված AdaBoost համակարգի արդյունքներ և նվաճումներ

Չորրորդ գլուխը ներկայացնում է հետազոտության կիրառական մասը՝ մեր մշակված MPI-հիմնված բաշխված AdaBoost համակարգի մանրամասն նկարագրությունը, մաթեմատիկական ֆորմալիզացիան, բարդության վերլուծությունը և համապարփակ փորձարարական գնահատումը: Այս գլխի հիմնական նպատակն է ապացուցել առաջարկվող մեթոդի տեսական հիմնավորվածությունը և գործնական արդյունավետությունը իրական պայմաններում:

Ճարտարապետական նվաճումներ և նորարարություններ

Մշակված համակարգի ճարտարապետությունը հիմնված է երեք հիմնական նորարարական բաղադրիչների վրա, որոնք իրենց համակցությամբ ապահովում են որակապես նոր մակարդակի արդյունավետություն բաշխված boosting ալգորիթմների ոլորտում:

Առաջին նվաճումը կայանում է ադապտիվ հաղորդակցական ժամանակացույցի մեխանիզմի մշակման մեջ: Ավանդական բաշխված AdaBoost մեթոդները կիրառում են ֆիքսված ինտերվալներով համաժամեցում, ինչը հանգեցնում է հաղորդակցության ավելորդ ծախսերի կամ, հակառակը, անբավարար սինխրոնիզացիայի: Մեր ադապտիվ մեխանիզմը մոնիտորինգ է անում սխալի հարաբերակցության փոփոխությունը ($\Delta \epsilon$) և դիմամիկ կերպով կարգավորում համաժամեցման միջակայքները: Երբ $\Delta \epsilon < \theta_1$ (կայուն ուսուցման ինդիկատոր), միջակայքը մեծանում է՝ նվազեցնելով հաղորդակցության հաճախականությունը: Հակառակը, երբ $\Delta \epsilon > \theta_2$ (անկայունության ցուցիչ), միջակայքը նվազում է՝

ապահովելով ավելի հաճախակի սինխրոնիզացիա: Այս մեխանիզմը թույլ է տվել հասնել 35%-ով հաղորդակցային ծախսերի նվազման:

Երկրորդ նվաճումը վերաբերում է բուֆերային ռազմավարության նախագծմանը և իրականացմանը: Մեր բուֆերային մեխանիզմը կարող է տեղական մոդելները, սխալները և կշիռները կուտակել հատուկ բուֆերներում այն ժամանակահատվածներում, երբ համաժամեցումը չի գործարկվում: Այս կուտակված տվյալները ապագայում ինտեգրվում են հաջորդ համաժամեցման իրադարձության ժամանակ, ինչն ապահովում է, որ ոչ մի արժեքավոր թարմացում չի կորում: Բուֆերային ռազմավարությունը հատկապես արդյունավետ է անկայուն ցանցային պայմաններում, որտեղ ավանդական մեթոդները ենթարկվում են զգալի կատարողականության անկման:

Երրորդ նվաճումը կապված է MPI պրիմիտիվների օպտիմալացված օգտագործման հետ: Համակարգը օգտագործում է MPI Gather-ը տեղական մոդելների հավաքման, MPI Broadcast-ը գլոբալ մոդելի տարածման և MPI Allreduce-ը սխալների արագ ագրեգացիայի համար: Այս պրիմիտիվների ճիշտ կոմբինացիան և ժամանակացույցը ապահովում է հաղորդակցության նվազագույն overhead և մաքսիմալ նվազախմբության (throughput) ձեռքբերում:

Մաթեմատիկական ֆորմալիզացիա և տեսական հիմքեր

Ալգորիթմի մաթեմատիկական նկարագրությունը պահպանում է AdaBoost-ի դասական տեսական երաշխիքները՝ միաժամանակ ընդլայնելով դրանք բաշխված միջավայրի համար: Հիմնական բանաձևերն են.

Ադապտիվ միջակայքի կարգավորում:

$$I_{\square, i} = \{$$

$$I_{\square} + \alpha, \quad \text{երթ } \Delta \varepsilon_{\square} < \theta_1$$

$$\max(1, I_{\square} - \beta), \quad \text{երթ } \Delta \varepsilon_{\square} > \theta_2$$

$$I_{\square}, \quad \text{այլապես}$$

}

Սխալի փոփոխության հաշվարկ:

$$\Delta \varepsilon_{\square} = |\text{mean}(\text{local errors}) - \varepsilon_{\square-1}|$$

Այս ֆորմալիզացիան թույլ է տալիս տեսականորեն ապացուցել, որ մեր ալգորիթմը պահպանում է AdaBoost-ի կոնվերգենցիայի հատկությունները՝ միաժամանակ ապահովելով բաշխված իրականացման առավելությունները:

Բարդության վերլուծության հիմնական եզրահանգումներ

Բարդության վերլուծությունը բացահայտեց մի քանի կարևոր տեսական և գործնական արդյունք: Հաշվարկային բարդությունը կազմում է $O(n/P)$, որտեղ n -ը տվյալների ընդհանուր քանակն է, իսկ P -ը՝ հանգույցների թիվը: Սա ցույց է տալիս գծային մասշտաբավորելիություն տվյալների ծավալի նկատմամբ:

Հաղորդակցական բարդությունը ավանդական մեթոդներում կազմում է $O(PT)$ յուրաքանչյուր boosting փուլի համար, որտեղ T -ը փուլերի քանակն է: Մեր ադապտիվ մեխանիզմի շնորհիվ այս բարդությունը միջինում նվազում է 35%-ով, ինչն արտացոլված է փորձարարական արդյունքներում՝ որպես նշանակալի հաղորդակցային ծախսերի կրճատում:

Ալգորիթմի տարածական բարդությունը (memory complexity) նույնպես օպտիմալացված է բուֆերային մեխանիզմի շնորհիվ: Թեև մոմենտային հիշողության օգտագործումը մի փոքր ավելանում է բուֆերավորման պատճառով, ընդհանուր տարածական արդյունավետությունը բարելավվում է, քանի որ հանգույցները կարող են աշխատել ավելի մեծ տվյալային հավաքածուների հետ՝ առանց կենտրոնական հանգույցի բեռնվաճառության:

Փորձարարական գնահատման մեթոդաբանությունը նախագծվել է այնպես, որ ապահովի համապարփակ և օբյեկտիվ արդյունքների ստացում: Փորձարկումները իրականացվել են տարբեր կոնֆիգուրացիաներով՝ 1-ից 32 հանգույցներ, յուրաքանչյուրը Intel Xeon պրոցեսորներով՝ 2.4 GHz հաճախականությամբ, 64GB RAM և InfiniBand ցանցային կապով: Յուրաքանչյուր կոնֆիգուրացիան կրկնվել է առնվազն 5 անգամ վիճակագրական վստահելիության ապահովման նպատակով:

Հիմնական փորձարարական արդյունքներ

Ստացված արդյունքները հաստատում են առաջարկվող մեթոդի բազմակողմանի գերազանցությունը ինչպես դասական AdaBoost-ի, այնպես էլ գոյություն ունեցող բաշխված իրականացումների նկատմամբ:

Մասշտաբավորելիության վերլուծություն

Մասշտաբավորելիության ուսումնասիրությունները ցույց տվեցին, որ մեր ալգորիթմը ցուցաբերում է գրեթե իդեալական գծային մասշտաբավորելիություն մինչև 32 հանգույց: Աղյուսակային արդյունքները ցույց են տալիս.

1 հանգույց: 83% ճշգրտություն, 115 վայրկյան

8 հանգույց: 87% ճշգրտություն, 20 վայրկյան

32 հանգույց: 89% ճշգրտություն, 8 վայրկյան

Այս արդյունքները վկայում են, որ հանգույցների քանակի ավելացումը ոչ միայն արագացնում է հաշվարկը, այլև բարելավում մոդելի ճշգրտությունը՝ շնորհիվ ավելի լայն տվյալային բազայի արդյունավետ օգտագործման:

Ադապտիվ մեխանիզմների գործնական ազդեցություն

Ադապտիվ համաժամեցման մեխանիզմի գործնական ազդեցությունը հատկապես ակնառու է եղել անկայուն ցանցային պայմաններում: Փորձարկումները, որտեղ արհեստականորեն ստեղծվել են 20%-ով ցանցային ուշացումներ, ցույց տվեցին, որ մեր մեթոդը պահպանում է 95%-ից ավելի իր սկզբնական կատարողականությունը, մինչդեռ ավանդական մեթոդները գրանցում են 30-40% կատարողականության անկում:

ԲուՖերային մեխանիզմի արդյունավետությունը նույնպես հաստատվել է գործնական փորձարկումներում: Այն ապահովում է, որ կարճաժամկետ ցանցային խնդիրները կամ հանգույցների ժամանակավոր անհասանելիությունը չեն հանգեցնում տվյալների կորուստի կամ մոդելի ցնցման:

Համակարգի դիմադրողականություն և կայունություն

Առաջարկվող համակարգը ցուցաբերել է բարձր մակարդակի դիմադրողականություն տարբեր անսպասելի իրավիճակների նկատմամբ: Հանգույցներից մեկի պլանային կամ ապլանային անջատման դեպքում համակարգը ավտոմատ վերակառուցում է տվյալների բաշխումը և շարունակում նորմալ գործունեությունը՝ միջինում 10-15 վայրկյանի տերության մեջ:

Տեսական ներդրման գիտական արժեքը

Այս հետազոտության տեսական ներդրումը ոչ միայն գործնական, այլև գիտական տեսանկյունից է արժեքավոր: Error-rate-based adaptive synchronization-ի գաղափարը կարող է կիրառվել այլ բաշխված մեքենայական ուսուցման ալգորիթմներում, ինչպիսիք են բաշխված SVM-ները, neural network-ները և այլ ensemble մեթոդները:

ԲուՖերային ռազմավարության մեթոդաբանությունը նույնպես ունի լայն կիրառական հեռանկարներ, հատկապես edge computing և IoT միջավայրերում, որտեղ ցանցային կապը հաճախ անկայուն է:

Սահմանափակումներ և հետագա զարգացման ուղղություններ

Չնայած ստացված դրական արդյունքներին, հետազոտությունը ունի որոշակի սահմանափակումներ, որոնք բացում են հետագա հետազոտական աշխատանքների հեռանկարները:

Ալգորիթմի ներկայիս իրականացումը առավելապես կենտրոնացած է դասակարգման խնդիրների վրա: Ռեգրեսիայի և այլ տիպի վերահսկվող ուսուցման խնդիրների համար անհրաժեշտ են լրացուցիչ հարմարեցումներ և օպտիմալացումներ:

Ադապտիվ պարամետրերի (θ_1 , θ_2 , α , β) ինքնակարգավորումը կարող է հետագայում ավտոմատացվել մեթա-ուսուցման (meta-learning) տեխնիկաների միջոցով, ինչը կբարելավի համակարգի ինքնազարգացման ունակությունները:

Եզրահանգումներ

Չորրորդ գլխի համապարփակ վերլուծությունը հստակ ապացուցում է առաջարկվող MPI-հիմնված բաշխված AdaBoost ալգորիթմի գիտական և գործնական արժեքը: Ստեղծված համակարգը հանդիսանում է զգալի քայլ առաջ բաշխված boosting ալգորիթմների ոլորտում՝ համատեղելով տեսական գրողականություն, ինժեներական նորարարություն և գործնական արդյունավետություն:

Ալգորիթմի եռակողմանի նվաճումը՝ ադապտիվ հաղորդակցային ժամանակացույց, բուֆերային ռազմավարություն և MPI պրիմիտիվների օպտիմալացված օգտագործում, ստեղծում է գույակական նոր ստանդարտ բաշխված ensemble ուսուցման համար: Ստացված 35%-ով հաղորդակցային ծախսերի նվազումը, 32%-ով ժամանակային կրճատումը և 1.2%-ով ճշգրտության բարելավումը գործնականում նշանակում են, որ համակարգը կարող է արդյունավետ մշակել ավելի մեծ տվյալային բազմություններ, ավելի քիչ ռեսուրսներով և ավելի բարձր որակով:

Այս արդյունքները բացում են նոր հեռանկարներ ինչպես ակադեմիական հետազոտությունների, այնպես էլ արդյունաբերական կիրառությունների համար, հատկապես այն ոլորտներում, որտեղ անհրաժեշտ է մշակել մեծածավալ տվյալներ սահմանված ժամանակահատվածում և ռեսուրսային սահմանափակումներով: