به نام خدا

راهنما و مجموعه خلاصه مقالات و سخنرانی‌های

پنجمین کنفرانس ملی انفورماتیک ایران

پژوهشکده علوم کامپیوتر
پژوهشگاه دانشهای بنیادی
پردیس فرمانیه، تهران

۱۳ و ۱۴ دی‌ماه ۱۴۰۲

برگزارکنندگان

پژوهشگاه دانش‌های بنیادی

انجمن انفورماتیک ایران

حامیان

شرکت ملی انفورماتیک
National Informatics Corporation

شرکت پویا

فهرست مطالب

پیام دبیران کنفرانس

با کمال خوشوقتی برگزاری پنجمین کنفرانس ملی انفورماتیک ایران را طی روزهای ۱۳ و ۱۴ دی ماه ۱۴۰۲ به اطلاع می‌رسانیم. این کنفرانس با همت و همکاری پژوهشگاه دانشهای بنیادی و انجمن انفورماتیک ایران برگزار می‌گردد. هدف اصلی از برگزاری این رویداد، گردهم‌آوری متخصصان، پژوهشگران و صنعتگران مرتبط با این حوزه مهم از علوم و مهندسی می‌باشد تا در محیطی علمی و تخصصی به تبادل مسائل و یافته‌های مهم خود بپردازند.

در این کنفرانس، بر اساس دسته‌بندی مرسوم در انجمن‌ها و محافل تخصصی انفورماتیک، مقالات در چهار شاخه ۱) سیستم، ۲) تئوری، ۳) هوش مصنوعی و ۴) زمینه های بین‌رشته‌ای، ارسال و داوری شدند. برای هر یک از مقالات ارسالی در هر شاخه، داوری حرفه‌ای با حداقل سه داور انجام شد و نهایتا ۱۹ مقاله که از کیفیت بسیار بالایی برخوردار بودند برای ارائه شفاهی در کنفرانس پذیرفته شدند.

در کنار ارائه مقالات عادی کنفرانس، کمیته علمی کنفرانس اقدام به دعوت از دانشمندان و پژوهشگران فعال و شناخته شده داخل و خارج کشور در این حوزه برای ارائه سخنرانیهای علمی کرد. خوشوقتیم به اطلاع برسانیم که در نتیجه این اقدام، ۹ سخنرانی علمی بسیار جذاب توسط پژوهشگران فعال در حوزه انفورماتیک در برنامه کنفرانس گنجانده شد. همچنین، علاوه بر سخنرانی‌های در نظر گرفته شده، دو میزگرد در خصوص هوش مصنوعی در بانکداری الکترونیکی ایران و چالش‌های پژوهش و صنعت در محاسبات لبه‌ای با حضور افراد شناخته شده در این حوزه خواهیم داشت.

بلافاصله پس از برگزاری جلسات کنفرانس در روزهای ۱۳ و ۱۴ دی ماه، در روز جمعه ۱۵ دی ماه، ۳ کارگاه آموزشی با موضوع‌های متنوع و جذاب، توسط دانشمندان و متخصصان متبحر حوزه انفورماتیک در دو نوبت صبح و بعدازظهر برگزار خواهد شد.

مراتب سپاس خود را از سخنران افتتاحیه سرکار خانم دکتر مرجان سیرجانی از دانشگاه Malardalen سوئد که دعوت کنفرانس را پذیرفتند تا در افتتاحیه کنفرانس به ایراد سخنرانی بپردازند ابراز می‌کنیم. همچنین از سرکار خانم دکتر صحرا صدیق سروستانی از دانشگاه Missouri، آقای دکتر امیرحسین جهانگیر از دانشگاه صنعتی شریف، آقای دکتر سعید صالحی از دانشگاه تبریز، آقای دکتر شهریار ابراهیمی از پژوهشگاه NCBR-IDEAS، آقای دکتر حسین حجت، از دانشگاه TeIAS و تهران، آقای دکتر جاوید طاهری از دانشگاه Karlstad، آقای دکتر صادق طالبی از دانشگاه Copenhagen، و آقای دکتر صدرالساداتی از دانشگاه ETH که به دعوت کنفرانس پاسخ مثبت دادند و با ایراد سخنرانی، جدیدترین یافته‌های پژوهشی خود را ارائه دادند کمال تشکر و قدردانی را می‌نماییم.

همچنین لازم می‌دانیم از دکتر محمدعلی اخایی، دکتر الهام فراهانی، دکتر کمال‌الدین یعقوبی رفیع، دکتر مهدی راستی، دکتر رضا طحان، و دکتر احمد خونساری که با شرکت در میزگرد به هرچه پربارتر شدن این رویداد کمک کردند، تشکر و قدردانی نماییم.

مایلیم از اعضای کمیته‌های مختلف کنفرانس و بخصوص کمیته علمی، مسئولین شاخه‌های آن و داوران مقالات که با دقت بسیار به بررسی مقالات و تشکیل برنامه کنفرانس پرداختند تشکر کنیم و قدردان نویسندگان مقالات ارسال شده، چه آنان که پذیرفته شدند و چه آنان که پذیرفته نشدند، می‌باشیم.

همچنین از حامیان کنفرانس، شرکت ملی انفورماتیک و شرکت پویا، که با حمایت مالی خود در برگزاری مناسب کنفرانس ما را یاری کردند تشکر می‌کنیم.

در انتها از تمامی عزیزانی که به هر نحو ممکن ما را در برگزاری کنفرانس یاری کردند و بخصوص شما شرکت کنندگان در جلسات مختلف کنفرانس و کارگاه‌ها تشکر می‌کنیم و امیدواریم برنامه‌های این کنفرانس مورد استفاده و توجه شما قرار گیرد.

دکتر پژمان لطفی کامران، دبیر کنفرانس دکتر احمد خونساری، دبیر کمیته علمی

برنامه‌ی کنفرانس در یک نگاه

روز اول: چهارشنبه ۱۳ دی‌ماه ۱۴۰۲	
۹:۱۵-۹:۰۰	افتتاحیه کنفرانس
۱۰:۰۰-۹:۱۵	سخنرانی افتتاحیه
۱۰:۱۵-۱۰:۰۰	استراحت
۱۱:۱۵-۱۰:۱۵	ارایه مقالات علمی گروه سیستم
۱۱:۳۰-۱۱:۱۵	استراحت
۱۲:۱۵-۱۱:۳۰	سخنرانی کلیدی ۲
۱۳:۰۰-۱۲:۱۵	استراحت
۱۳:۴۵-۱۳:۰۰	سخنرانی کلیدی ۳
۱۳:۵۰-۱۳:۴۵	استراحت
۱۵:۱۰-۱۳:۵۰	میزگرد ۱
۱۵:۱۵-۱۵:۱۰	استراحت
۱۶:۰۰-۱۵:۱۵	سخنرانی کلیدی ۴
۱۶:۱۰-۱۶:۰۰	استراحت
۱۶:۴۰-۱۶:۱۰	ارائه مقالات علمی گروه هوش مصنوعی
۱۷:۰۰-۱۶:۴۰	پژوهش دانشجویان دکتری
۱۷:۴۵-۱۷:۰۰	سخنرانی کلیدی ۵
۱۸:۳۰-۱۷:۴۵	ارائه مقالات علمی گروه سیستم
روز دوم: پنج‌شنبه ۱۴ دی‌ماه ۱۴۰۲	

۹:۰۰-۹:۴۵	سخنرانی کلیدی ۶
۹:۴۵-۱۰:۰۰	استراحت
۱۰:۰۰-۱۱:۰۰	ارائه مقالات علمی گروه هوش مصنوعی/ سیستم
۱۱:۰۰-۱۱:۱۵	استراحت
۱۱:۱۵-۱۲:۰۰	سخنرانی کلیدی ۷
۱۳:۰۰-۱۲:۰۰	استراحت
۱۳:۰۰-۱۳:۴۵	سخنرانی کلیدی ۸
۱۳:۴۵-۱۴:۰۰	استراحت
۱۴:۰۰-۱۵:۱۵	ارایه مقالات علمی گروه زمینه‌های بین رشته‌ای
۱۵:۳۰-۱۵:۱۵	استراحت
۱۶:۴۵-۱۵:۳۰	میزگرد ۲
۱۷:۰۰-۱۶:۴۵	استراحت
۱۷:۴۵-۱۷:۰۰	سخنرانی کلیدی ۹
۱۸:۰۰-۱۷:۴۵	اختتامیه کنفرانس

برگزارکنندگان کنفرانس

روسای کنفرانس: محمدجواد اردشیر لاریجانی، رئیس پژوهشگاه دانش‌های بنیادی

ابراهیم نقیب‌زاده مشایخ، رئیس انجمن انفورماتیک ایران

دبیر کنفرانس: پژمان لطفی‌کامران، پژوهشگاه دانش‌های بنیادی

دبیر کمیته علمی: احمد خونساری، دانشگاه تهران و پژوهشگاه دانش‌های بنیادی

دبیران شاخه‌های علمی:

دکتر مهدی مدرسی، دانشگاه تهران (شاخه سیستم)
دکتر محسن ابراهیمی مقدم، دانشگاه شهید بهشتی (شاخه هوش مصنوعی)
دکتر حمید بیگی، دانشگاه صنعتی شریف (شاخه تئوری)
دکتر محمد عبدالهی‌ازگمی، دانشگاه علم و صنعت (شاخه زمینه‌های بین‌رشته‌ای)

کمیته اجرایی:

زهرا رضوانی و سید محمد حسینی، پژوهشگاه دانش‌های بنیادی (تبلیغات)
نظام رهبانی، پژوهشگاه دانش‌های بنیادی (اینترنت)
مهدی دولتی، دانشگاه علم و صنعت ایران (میزگردها)
سمیرا حسین‌قربان و فاطمه بهاری‌فرد ، پژوهشگاه دانش‌های بنیادی (ارتباط با صنعت و امور حامیان)
نعیمه امیدوار، پژوهشگاه دانش‌های بنیادی (انتشارات)
نعیمه امیدوار، پژوهشگاه دانش‌های بنیادی (پژوهش دانشجویان دکترا)
مهدی دولتی، پژوهشگاه دانش‌های بنیادی (ارتباط با پایگاه استنادی جهان اسلام)
سینا دارابی‌مقدم و سپیده صفری، پژوهشگاه دانش‌های بنیادی (کارگاه‌ها)
فاطمه آقایی‌پور و سید حامد رستگار، پژوهشگاه دانش‌های بنیادی (سامانه مجازی)
حمیدرضا شهرابی‌فراهانی، پژوهشگاه دانش‌های بنیادی (امور اجرایی)

اعضای کمیته علمی

اعضای کمیته علمی شاخه‌ی سیستم:

- مصطفی ارسالی صالحی نسب (دانشگاه تهران)
- محسن انصاری (دانشگاه صنعتی شریف)
- امیر امینی‌فر (دانشگاه Lund- سوئد)
- سیاوش بیات سرمدی (دانشگاه صنعتی شریف)
- حاکم بیت‌اللهی (دانشگاه علم و صنعت)
- فرهاد پاکدامن (دانشگاه تهران)
- سید محمد حسینی (پژوهشگاه دانش‌های بنیادی)
- مسعود دانش‌طلب (دانشگاه mdh-سوئد)
- مهدی دولتی (پژوهشگاه دانش‌های بنیادی)
- مسعود دهیادگاری (دانشگاه صنعتی خواجه نصیر)
- دارا رحمتی (دانشگاه شهید بهشتی)
- مهدی رضایی (پژوهشگاه دانش‌های بنیادی)
- نظام رهبانی (پژوهشگاه دانش‌های بنیادی)
- حمیدرضا زرندی (دانشگاه صنعتی امیرکبیر)
- سیما سینایی (دانشگاه mdh- سوئد)
- حسین شفیعی (دانشگاه صنعتی خواجه‌نصیرالدین طوسی)
- سپیده صفری (پژوهشگاه دانش‌های بنیادی)
- حامد فربه (دانشگاه صنعتی امیرکبیر)
- هاجر فلاحتی (پژوهشگاه دانش‌های بنیادی)
- مهدی کمال (دانشگاه تهران)
- سعید گرگین (سازمان پژوهش‌های علمی و صنعتی ایران)
- پرهام مرادی (دانشگاه کردستان)
- محمود نادران طهان (دانشگاه شهید چمران اهواز)
- سینا دارابی مقدم (پژوهشگاه دانش‌های بنیادی)

اعضای کمیته علمی شاخه‌ی هوش مصنوعی:

- فاطمه آقایی‌پور (پژوهشگاه دانش‌های بنیادی)
- نعیمه امیدوار (پژوهشگاه دانش‌های بنیادی)
- سعید بیگدلی (شرکت ایزایران)
- زهرا رضوانی (پژوهشگاه دانش‌های بنیادی)

- علیرضا رضوانیان (دانشگاه علم و فرهنگ)
- زهرا ریاحی (دانشگاه پنسیلوانیا)
- محمد سبک‌رو (پژوهشگاه دانش‌های بنیادی)
- یاسر شکفته (دانشگاه شهید بهشتی)
- مهرنوش شمس فرد (دانشگاه شهید بهشتی)
- احمد علی آبین (دانشگاه شهید بهشتی)
- محمد مهدی فقیه (دانشگاه صنعتی کرمان)
- هشام فیلی (دانشگاه تهران)
- حامد ملک (دانشگاه شهید بهشتی)
- آرمین سلیمی بدر (دانشگاه شهید بهشتی)
- رضا خسروآبادی (دانشگاه شهید بهشتی)
- علیرضا طالب‌پور (دانشگاه شهید بهشتی)
- محسن سریانی (دانشگاه علم و صنعت ایران)
- منصور جم‌زاد (دانشگاه صنعتی شریف)
- حسام عمران‌پور (دانشگاه نوشیروانی)

اعضای کمیته علمی شاخه‌ی تئوری:

- محمدعلی آبام (دانشگاه صنعتی شریف)
- آرش احدی (دانشگاه خوارزمی)
- ابراهیم اردشیر لاریجانی (دانشگاه علم و صنعت ایران)
- محمد ایزدی (دانشگاه صنعتی شریف)
- علیرضا باقری (دانشگاه صنعتی امیرکبیر)
- فاطمه بهاری‌فرد (پژوهشگاه دانش‌های بنیادی)
- سمیرا حسین‌قربان (پژوهشگاه دانش‌های بنیادی)
- منصور داودی منفرد (دانشگاه تحصیلات تکمیلی علوم پایه زنجان)
- زاهد رحمتی (دانشگاه صنعتی امیرکبیر)
- مسعود صدیقین (دانشگاه خاتم)
- حمید ضرابی‌زاده (دانشگاه صنعتی شریف)
- مریم طهماسبی آبدر (دانشگاه شهید بهشتی)
- شراره علیپور (دانشگاه خاتم)
- علی غلامی رودی (دانشگاه صنعتی نوشیروانی بابل)
- امین غیبی (دانشگاه صنعتی امیرکبیر) محمد فرشی (دانشگاه یزد)
- محمد هادی فروغمند اعرابی (دانشگاه صنعتی شریف)
- محمدمهدی کیخا (دانشگاه سیستان و بلوچستان)
- احمد مرادی (دانشگاه مازندران)
- معصومه مرادیان (پژوهشگاه دانش‌های بنیادی)
- سلما سادات مهدوی (پژوهشگاه دانش‌های بنیادی)

- مصطفی نوری بایگی (دانشگاه فردوسی مشهد)
- زهرا نیلفروشان (دانشگاه خوارزمی)
- مهدیه هاشمی نژاد (دانشگاه یزد)

اعضای کمیته علمی شاخه‌ی زمینه‌های بین رشته‌ای:

- مهرداد آشتیانی (دانشگاه علم و صنعت ایران)
- محمدرضا ابراهیمی دیشابی (دانشگاه آزاد اسلامی)
- جعفر الماسی‌زاده (دانشگاه اصفهان)
- رضا انتظاری ملکی (دانشگاه علم و صنعت ایران)
- علیرضا باقری (دانشگاه صنعتی امیرکبیر)
- سوده حسینی (دانشگاه شهید باهنر کرمان)
- سید حامد رستگار (پژوهشگاه دانش‌های بنیادی)
- بهمن زمانی (دانشگاه اصفهان)
- حامد سپهرزاده (دانشکده فنی و حرفه‌ای شهید شمسی‌پور)
- سعید صدیقیان کاشی (دانشگاه خواجه نصیرالدین طوسی)
- مهدی کارگهی (دانشگاه تهران)
- احمد محمودی ازناوه (دانشگاه شهید بهشتی)
- حسن مطلبی پاقلعه (دانشگاه تحصیلات تکمیلی صنعتی کرمان)
- علی نقاش اسدی (دانشکده فنی فومن، دانشگاه تهران)
- امیرمهدی سازدار (دانشگاه علوم و فنون هوایی شهید ستاری)
- عبدالله غفاری ششجوانی (دانشگاه علوم و فنون هوایی شهید ستاری)
- مهدی دولتی (دانشگاه صنعتی شریف)

زمینه‌های علمی کنفرانس

امسال مقالات کنفرانس ملی انفورماتیک ایران در چهار شاخه اصلی زیر سازمان‌دهی شده اند:

۱- سیستم
- معماری کامپیوتر
- سیستم حافظه و ذخیره‌سازی داده
- شبکه کامپیوتری
- امنیت داده و کامپیوتر
- پایگاه داده
- سیستم‌های تعبیه شده و کم‌توان
- سیستم‌های بی‌درنگ
- پردازش با کارایی بالا
- ارزیابی و تحلیل کارایی
- سیستم‌های عامل
- زبان‌های برنامه‌سازی
- مهندسی نرم افزار
- پردازش موبایل
- رایانش ابری
- دیگر موضوع‌های مرتبط با این شاخه

۲- هوش مصنوعی
- هوش مصنوعی
- یادگیری ماشین
- رایانش نرم
- شناسایی الگو
- داده‌کاوی
- پردازش زبان طبیعی
- بینایی ماشین و پردازش تصویر
- وب و بازیابی اطلاعات
- دیگر موضوع‌های مرتبط با این شاخه

۳- تئوری
- طراحی و تحلیل الگوریتم‌ها
- هندسه محاسباتی
- الگوریتم‌های تقریبی و تصادفی
- پیچیدگی محاسبات
- الگوریتم‌های کوانتومی
- الگوریتم‌های موازی و توزیع‌شده
- منطق و اعتبارسنجی
- روش‌های صوری
- دیگر موضوع‌های مرتبط با این شاخه

۴- زمینه‌های بین‌رشته‌ای
- پردازش داده‌های کلان
- بیوانفورماتیک
- گرافیک کامپیوتری
- اقتصاد و محاسبات
- تعامل انسان و کامپیوتر
- رباتیک
- مصورسازی
- اینترنت اشیا
- سایر موضوع‌های بین‌رشته‌ای

سخنرانیهای علمی چهارمین دوره‌ی کنفرانس ملی انفورماتیک ایران

دکتر مرجان سیرجانی
دانشگاه
Malardalen

دکتر صحرا صدیق سروستانی
دانشگاه Missouri

دکتر امیرحسین جهانگیر
دانشگاه صنعتی شریف

دکتر سعید صالحی
دانشگاه تبریز

دکتر شهریار ابراهیمی
پژوهشگاه IDEAS-NCBR

دکتر حسین حجت
دانشگاه TeIAS و تهران

دکتر جاوید طاهری
دانشگاه Karlstad

آقای دکتر صادق طالبی
دانشگاه Copenhagen

دکتر محمد صدرالساداتی
دانشگاه ETH

برنامه‌ی پنجمین کنفرانس ملی انفورماتیک ایران

روز اول: چهارشنبه ۱۳ دی ماه ۱۴۰۲

	زمان	برنامه کنفرانس	رئیس نشست
افتتاحیه	۹:۰۰ – ۹:۱۵	خوش‌آمدگویی مسئولین کنفرانس	
سخنرانی افتتاحیه	۹:۱۵ – ۱۰:۰۰	سخنران: خانم دکتر مرجان سیرجانی، دانشگاه Malardalen، سوئد عنوان: Timed actors and their formal verification	آقای دکتر احمد خونساری و آقای دکتر پژمان لطفی کامران
استراحت	۱۰:۰۰ – ۱۰:۱۵		
مقالات علمی گروه سیستم	۱۰:۱۵ – ۱۰:۳۰	پردازش بهینه مرحله آموزش شبکه عصبی با استفاده از اشتراک داده زهرا رحیمی، هاجر فلاحتی، حاکم بیت الهی	خانم دکتر پریا دربانی
	۱۰:۳۰ – ۱۰:۴۵	ارائه طرح تشویقی و اولویت‌بندی وظایف جهت استفاده بهینه از انرژی مازاد خودروهای الکتریکی در محاسبات مه به کمک کنترل‌کننده SDN فائزه رحمانی، نیک محمد بلوچزهی	
	۱۰:۴۵ – ۱۱:۰۰	مدیریت منابع مبتنی بر نظریه بازی برای کاربردهای بی‌درنگ با استفاده از Lévy Walk در سامانه‌های لبه ابوالفضل یونسی، محسن انصاری	
	۱۱:۰۰ – ۱۱:۱۵	زمانبندی آگاه به اوج توان در سامانه‌های بحرانی-مختلط سه سطحی چندهسته‌ای شایان شکری، محراب طوقانی، ساره ملکی، سپیده صفری، شاهین حسابی	
استراحت	۱۱:۱۵ – ۱۱:۳۰		
سخنرانی ۲	۱۱:۳۰ – ۱۲:۱۵	سخنران: خانم دکتر صحرا صدیق سروستانی، دانشگاه Missouri، امریکا عنوان: Links, sequences, and consequences – a bird’s eye view of dependability and security of complex networked systems	خانم دکتر نعیمه امیدوار
استراحت	۱۲:۱۵ – ۱۳:۰۰		
سخنرانی ۳	۱۳:۰۰ – ۱۳:۴۵	سخنران: آقای دکتر امیرحسین جهانگیر، دانشگاه صنعتی شریف، ایران عنوان: چرا پردازنده‌های پیشرفته امروزی دستورات را همان‌گونه که برنامه می‌نویسیم اجرا نمی‌کنند؟ آیا زمان تعریف یک مدل محاسباتی جدید فرا نرسیده است؟	آقای دکتر اسلام ناظمی
استراحت	۱۳:۴۵ – ۱۳:۵۰		
میزگرد ۱	۱۳:۵۰ – ۱۵:۱۰	موضوع: هوش مصنوعی در بانکداری الکترونیکی ایران آقای دکتر محمدعلی اخایی، عضو هیات علمی دانشگاه تهران، پردازش زبان طبیعی در بانکداری هوشمند؛ نیازمندی‌ها، راه حل‌ها. خانم دکتر الهام فراهانی، مدرس دانشگاه صنعتی شریف، عضو هیات مدیره انجمن کامپیوتر ایران، متاورس و بانکداری هوشمند. آقای دکتر کمال‌الدین یعقوبی رفیع، شرکت ملی انفورماتیک، پیشنیازهای حرکت به سمت بانکداری هوشمند. رئیس نشست: آقای دکتر اسلام ناظمی، نایب رئیس هیات مدیره انجمن انفورماتیک ایران	آقای دکتر اسلام ناظمی
استراحت	۱۵:۱۰ – ۱۵:۱۵		
سخنرانی ۴	۱۵:۱۵ – ۱۶:۰۰	سخنران: آقای دکتر سعید صالحی، دانشگاه تبریز، ایران عنوان: On the Halting Probability and Chaitin's Heuristic Principle	آقای دکتر اسلام ناظمی
استراحت	۱۶:۰۰ – ۱۶:۱۰		
مقالات علمی گروه هوش مصنوعی	۱۶:۱۰ – ۱۶:۲۵	بازیابی تصاویر محتوامحور با استفاده از ویژگی‌های بافت استخراج شده از الگوی باینری محلی دو لایه سید علی حسینی، امیرحسین عشقی، صبا محمدی	آقای دکتر اسلام ناظمی
	۱۶:۲۵ – ۱۶:۴۰	الگوریتم ژنتیک چند هدفه مرتب‌سازی نامغلوب مبتنی بر خوشه‌بندی فازی پژمان غلام نژاد، امیرمهدی سازدار، عبدالله غفاری	
پژوهش دانشجویان دکتری	۱۶:۴۰ – ۱۷:۰۰	پیش‌بینی تفسیرپذیر نتیجه‌ی فرایند کسب‌وکار زهرا حسینی نژاد محبتی، صادق علی اکبری، رامک قوامی زاده میبدی، حامد ملک	آقای دکتر اسلام ناظمی
سخنرانی ۵	۱۷:۰۰ – ۱۷:۴۵	سخنران: آقای دکتر شهریار ابراهیمی از پژوهشگاه IDEAS-NCBR ، لهستان عنوان: Zero-Knowledge Proofs in Action	آقای دکتر آرش واعظی
مقالات علمی گروه سیستم	۱۷:۴۵ – ۱۸:۰۰	روش پیش‌واکشی داده کارا در پردازنده‌های گرافیکی صبا مستوفی، هاجر فلاحتی، نگین ماهانی، پژمان لطفی کامران، حمید سربازی آزاد	خانم دکتر زهرا رضوانی
	۱۸:۰۰ – ۱۸:۱۵	پیش‌بینی فعالیت منجر به گلوگاه در فرایندهای کسب‌وکار با استفاده از روش‌های فرایندکاوی زهرا حسینی نژادمحبتی، صادق علی‌اکبری، معصومه کوهستانی	
	۱۸:۱۵ – ۱۸:۳۰	شتابدهنده مبتنی بر پردازش درون حافظه برای شبکه های عصبی ژرف هاجر فلاحتی، نگین ماهانی	

برنامه‌ی پنجمین کنفرانس ملی انفورماتیک ایران

روز دوم: پنجشنبه ۱۴ دی ماه ۱۴۰۲

	زمان	برنامه کنفرانس	رئیس نشست
سخنرانی ۶	۹:۰۰ – ۹:۴۵	سخنران: آقای دکتر حسین حجت، دانشگاه TeIAS و تهران، ایران عنوان: Programming Abstractions for Networks	خانم دکتر سمیرا حسین قربان
استراحت	۹:۴۵ – ۱۰:۰۰		
مقالات علمی گروه هوش مصنوعی/ سیستم	۱۰:۰۰ – ۱۰:۱۵	بیشینه‌سازی انتشار در شبکه‌های اجتماعی با استفاده از الگوریتم ژنتیک محسن قنبری قمصری، سید مهدی وحیدی پور، فرشته دهقانی	آقای دکتر محمد گنج تابش
	۱۰:۱۵ – ۱۰:۳۰	پیش‌بینی بیماری‌های مزمن با داده‌های نامتوازن توسط ماشین بردار پشتیبان گرانشی عبدالله محمدی، جلال الدین نصیری، سهراب عفتی	
	۱۰:۳۰ – ۱۰:۴۵	پیش‌بینی ابتلا به بیماری‌های مزمن به کمک ماشین بردار پشتیبان دوقلو با قیود نرم حمیده فدیشه‌ای، جلال الدین نصیری، سهراب عفتی	
	۱۰:۴۵ – ۱۱:۰۰	بهبود گراف زوم؛ چارچوبی برای تحلیل بازنمایی گراف سید مهدی وحیدی‌پور، عماد صلاتی، رسول سبزه‌واری، محمد ریاضی	
استراحت	۱۱:۰۰ – ۱۱:۱۵		
سخنرانی ۷	۱۱:۱۵ – ۱۲:۰۰	سخنران: آقای دکتر جاوید طاهری، دانشگاه Karlstad، سوئد عنوان: Edge Intelligence	آقای دکتر محمد گنج تابش
استراحت	۱۲:۰۰ – ۱۳:۰۰		
سخنرانی ۸	۱۳:۰۰ – ۱۳:۴۵	سخنران: آقای دکتر صادق طالبی، دانشگاه Copenhagen، دانمارک عنوان: Exploration in reward machines with near-optimal regret	خانم دکتر فاطمه آقائی‌پور
استراحت	۱۳:۴۵ – ۱۴:۰۰		
مقالات علمی گروه زمینه‌های بین رشته‌ای	۱۴:۰۰ – ۱۴:۱۵	اقدامات امنیتی برای مقابله با تهدیدات امنیتی در برنامه‌های کاربردی اینترنت اشیا صدیقه هدایتی، پیام محمودی نصر	خانم دکتر سپیده صفری
	۱۴:۱۵ – ۱۴:۳۰	طبقه‌بندی سیگنال‌های قلبی توسط شبکه‌های عصبی convolutional وSqueezeNet سیده محبوبه مولوی عربشاهی، فاطمه معاون	
	۱۴:۳۰ – ۱۴:۴۵	یک چارچوب انتخاب ویژگی مرکب مبتنی بر معیار حداقل افزونگی و حداکثر ارتباط برای طبقه‌بندی داده‌های بیولوژیکی فاطمه کوکب زاده، الهام عباسی هرفته، جمال زارعپور احمدآبادی	
	۱۴:۴۵ – ۱۵:۰۰	بهبود عملکرد یافتن نوع مشتری با رویکرد چندمرحله‌ای در صنعت هتلداری حامد شرافت مولا، هادی یعقوبیان، راضیه ملک حسینی، کرم الله باقری فرد	
	۱۵:۰۰ – ۱۵:۱۵	سامانه سلامت‌سنجی حسگرهای درون خودرویی مبتنی بر شبکه عصبی خودرمزگذار و رگرسیون جنگل تصادفی: نمونه موردی سایپا سحر ترک حصاری، بهنام یوسفی مهر، مهدی قطعی	
استراحت	۱۵:۱۵ – ۱۵:۳۰		
میزگرد ۲	۱۵:۳۰ – ۱۶:۴۵	محاسبات لبه‌ای: چالش‌های پژوهش و صنعت دکتر مهدی راستی، دانشیار، دانشگاه اولو، فنلاند دکتر رضا طحان، مدیر سرویس‌های نوین شبکه، همراه اول دکتر احمد خونساری، دانشیار، دانشگاه تهران	آقای دکتر مهدی دولتی
استراحت	۱۶:۴۵-:۱۷:۰۰		
سخنرانی ۹	۱۷:۰۰-۱۷:۴۵	سخنران: آقای دکتر صدرالساداتی، دانشگاه ETH، سوئیس عنوان: Storage-Centric Computing	آقای دکتر اسلام ناظمی
اختتامیه کنفرانس	۱۷:۴۵-۱۸:۰۰		آقای دکتر اسلام ناظمی

برنامه‌ی کارگاه‌های پنجمین کنفرانس ملی انفورماتیک ایران

روز سوم: جمعه ۱۵ دی ماه ۱۴۰۲

زمان	برنامه کارگاه	رئیس نشست
کارگاه اول	کارگاه بلاکچین و ارز دیجیتال، کلیات حوزه‌های بلاکچین و کدنویسی آن برگزارکننده: ApexChain	خانم دکتر هاجر فلاحتی
۸:۰۰ – ۹:۳۰	۱. مفاهیم اولیه بلاکچین ۲. بلاکچین در عمل چیست و ارتباط آن با هوش مصنوعی در کجاست؟ ۳. بررسی حوزه‌های بلاکچین ۴. جایگاه توکن، ارزدیجیتال و بلاکچین ۵. حوزه ترید ارزدیجیتال	
۹:۳۰ – ۱۰:۰۰	استراحت	
۱۰:۰۰ – ۱۱:۳۰	۶. مروری بر دانش ترید ارزدیجیتال ۷. فرصت های سرمایه گذاری ارزدجیتال ۸. ماینینگ، امکانات ماینینگ و فرصت‌ها و هزینه‌ها ۹. بررسی مزایای تکنولوژی بلاکچین ۱۰. انواع زبان برنامه‌نویسی و پلتفورم های بلاکچین و ارزدیجیتال	
۱۱:۳۰ – ۱۲:۰۰	استراحت	

زمان	برنامه کارگاه	رئیس نشست
کارگاه دوم	Server bechmarking with CloudSuite 4.0 برگزارکننده: آقای علی انصاری	آقای دکتر سینا دارابی
۱۲:۰۰ – ۱۳:۳۰	CloudSuite is a benchmark suite for first-party cloud services. The suite consists of eight benchmarks representing popular online services and analytic workloads in datacenters. The benchmarks are based on state-of-the-art open-source real-world software stacks and are containerized for ease of use. Cloud computing is now the dominant platform to offer scalable online services to a global client base. Today's popular online services (e.g., web search, social networking, and business analytics) are characterized by massive working sets, deep software stacks, high degrees of request parallelism, and real-time constraints. These characteristics set cloud services apart from desktop (SPEC), parallel (PARSEC), and traditional commercial server workloads (TPC). Thus, we offer CloudSuite, enabling users to analyze their systems with representative cloud services. CloudSuite also complements emerging first-party workloads (e.g., Microservices) and third-party workloads (e.g., Serverless) with multi-tier monolithic software stacks that remain a backbone for datacenter services.	
۱۳:۳۰ – ۱۴:۰۰	استراحت	

زمان	برنامه کارگاه	رئیس نشست
کارگاه سوم	مقدمه ای بر یادگیری ماشین و کاربردها برگزارکننده: آقای هادیان دانشجوی دکترای دانشگاه بازل	خانم دکتر سلما سادات مهدوی
۱۴:۰۰ – ۱۵:۳۰	۱. مقدمه‌ای بر یادگیری ماشین ○ یادگیری با ناظر ○ یادگیری بدون ناظر ۲. مقدمه‌ای بر الگوریتم‌های طبقه بندی ○ درخت تصمیم ○ جنگل تصادفی	
۱۵:۳۰ – ۱۵:۴۵	استراحت	
۱۵:۴۵ – ۱۷:۱۵	۳. روش‌های ارزیابی مدل‌های طبقه بندی ۴. آشنایی با پکیج Scikit-learn در پایتون	

پنجمین کنفرانس ملی انفورماتیک ایران
پژوهشگاه دانشهای بنیادی، پردیس فرمانیه، تهران
۱۳ و ۱۴ دی ماه ۱۴۰۲

پیش‌بینی فعالیت منجر به گلوگاه در فرایندهای کسب‌وکار با استفاده از روش‌های فرایندکاوی

زهرا حسینی نژادمحبتی[1]، صادق علی‌اکبری[2]، معصومه کوهستانی[3]

[1] دانشکده مهندسی و علوم کامپیوتر، دانشگاه شهید بهشتی، تهران، ایران
z.hosseininezhad@sbu.ac.ir
[2] استادیار دانشکده مهندسی و علوم کامپیوتر، دانشگاه شهید بهشتی، تهران، ایران
s_aliakbary@sbu.ac.ir
[3] دانشکده مهندسی و علوم کامپیوتر، دانشگاه شهید بهشتی، تهران، ایران
ma.kouhestani@mail.sbu.ac.ir

چکیده

یکی از چالش‌های فرایندها، وجود گلوگاه است که روی عملکرد فرایند تاثیر می‌گذارد و به عنوان مثال باعث تاخیر در اجرای فرایند می‌شود. با کمک فرایندکاوی می‌توانیم به تحلیل گلوگاه‌ها بپردازیم. برای تحلیل گلوگاه می‌توان به کشف، پیش‌بینی و راهکارهای پیشگیری از وقوع گلوگاه‌ها اشاره کرد. در اکثر پژوهش‌های پیشین، در حوزه‌ی تحلیل گلوگاه با استفاده از روش‌های فرایندکاوی، صرفا کشف مورد توجه قرار گرفته است و مقالات کمتری به پیش‌بینی گلوگاه پرداخته‌اند و به‌ویژه، اکثر روش‌های موجود از یادگیری ماشین برای پیش‌بینی گلوگاه استفاده نکرده‌اند. در این مقاله ما به پیش‌بینی فعالیت‌هایی در یک فرایند که منجر به ایجاد تاخیر شده‌اند، می‌پردازیم؛ بدین منظور یک روش یادگیری گروهی ارائه کرده‌ایم. نتایج ارزیابی نشان می‌دهد این روش یادگیری گروهی منجر به بهبود معیارهای ارزیابی نسبت به روش‌های پایه شده است.

کلمات کلیدی

فرایند کسب‌وکار، فرایندکاوی، پیش‌بینی گلوگاه، یادگیری گروهی.

1- مقدمه

یک فرایند کسب‌وکار، مجموعه‌ای از رویدادها، فعالیت‌ها و نقاط تصمیم‌گیری است و شامل تعدادی نقش (انسان، سازمان یا سامانه‌ی نرم‌افزاری) و شیء (تجهیزات یا اسناد) است که نهایتا منجر به یک نتیجه‌ی ارزشمند برای مشتری می‌شود [1]. برای توصیف دقیق‌تر جریان کنترل در یک فرایند و نمایش آن، از مدل فرایند استفاده می‌شود که در حقیقت توصیف فرایند در قالب مجموعه‌ای از فعالیت‌ها و ترتیب اجرای آن‌ها است [2]. فرایندکاوی روش‌هایی برای کشف، پایش و بهبود فرایندهای کسب‌وکار از طریق استخراج دانش از تاریخچه‌ی اجرای فرایندها –که آن را نگاره رویداد می‌نامیم– فراهم می‌کند [2]. فرایندکاوی، روش‌ها و کاربردهای مختلفی دارد که یکی از آنها، تحلیل گلوگاه است. وجود گلوگاه در فرایندها، تاثیر زیادی روی عملکرد فرایند دارد. برای تحلیل گلوگاه، در اولین گام باید به دنبال تعریف دقیقی از گلوگاه باشیم. برای گلوگاه تعاریف گوناگونی ارائه شده است [4]. طبق یک تعریف، گلوگاه بر بازده سامانه اثر می‌گذارد و هر چه این تاثیر بیشتر باشد، اهمیت بیشتری دارد [6]. در تعریفی دیگر، گلوگاه یک محدودیت است و محدودیت یعنی هر آن چیزی که سامانه را از رسیدن به عملکرد بهتر باز می‌دارد [7]. در تعریف سوم، به منابع توجه می‌شود و براساس این تعریف، گلوگاه منبع یا منابعی است که کمترین ظرفیت را نسبت به سایر منابع دارند [7]. اما تعریف دیگری نیز از گلوگاه در مرجع [4] بیان شده که به تعریف اول (تاثیر بر بازده سامانه) مرتبط است و در این مقاله نیز این تعریف مدنظر قرار می‌گیرد. طبق این تعریف، گلوگاه یک یا مجموعه‌ای از فعالیت‌ها در سامانه است که کل فرایند را متوقف یا کند می‌کند. اگر این گلوگاه برطرف شود، عملکرد کلی سامانه بهبود یافته و باعث کاهش زمان و هزینه می‌شود.

در این مقاله، با کمک روش‌های یادگیری ماشین به پیش‌بینی فعالیت‌های منجر به گلوگاه خواهیم پرداخت. ساختار این مقاله در ادامه بدین صورت است: در بخش 2 تعاریف پایه مرور شده و مساله به شکل دقیق بیان می‌شود. سپس مروری کوتاه روی پژوهش‌های پیشین انجام می‌شود. بخش 3 به روش پیشنهادی و جزئیات آن می‌پردازد. بخش 4 به مجموعه داده و ارزیابی روش پیشنهادی اختصاص یافته است. نهایتا در بخش 5 نتیجه‌گیری و پیشنهاد کارهای آتی ارائه می‌شود.

2- پیش‌زمینه و کارهای مرتبط

2-1- تعاریف پایه

به یک نمونه‌ی اجرایی از فرایند، به اختصار «نمونه» گفته می‌شود. هر نمونه از تعدادی «رویداد» تشکیل می‌شود و این رویدادها بیانگر انجام «فعالیت»هایی هستند که توسط فرایند، مدل شده است. به مجموعه داده‌ای که شامل اطلاعات حاصل از اجرای نمونه‌های مختلف است، «نگاره رویداد» گفته می‌شود که ورودی اصلی در تحلیل‌های فرایندکاوی است [2]. در ادامه این مفاهیم را دقیق‌تر تعریف می‌کنیم:

رویداد: هر رویداد یک چندتایی است که در فرمول (1) نشان داده شده است. در این فرمول، a بیانگر نام فعالیت، c شناسه نمونه، t برچسب زمانی و (d_i, v_i)، ویژگی (d) و مقدار آن (v) برای سایر ویژگی‌ها (در صورت وجود) است [5].

$$e = (a, c, t, (d_1, v_1), \dots, (d_m, v_m)), m \geq 0 \quad (1)$$

دنباله: یک توالی غیرتهی از رویدادهای یک نمونه، دنباله نامیده می‌شود. فرمول (2) یک دنباله با طول n را تعریف می‌کند [5]. توجه به این نکته ضروری است که دنباله، شامل رویدادهایی است که همگی متعلق به یک نمونه اجرایی از فرایند هستند. بنابراین شناسه‌ی نمونه برای همگی رویدادهای این دنباله، یکسان است.

$$\sigma = \langle e_1, e_2, \dots, e_n \rangle, \quad \forall i, j \in [1, n]: e_i.c = e_j.c \quad (2)$$

نگاره رویداد: مجموعه‌ی تمام دنباله‌های کامل برای تمام نمونه‌های اجرایی یک فرایند، نگاره رویداد است که در فرمول (3) نشان داده شده است. منظور از دنباله‌ی کامل این است که این دنباله‌ها متعلق به اجرای کامل یک نمونه هستند، به عبارتی برای هر نمونه، تمام رویدادهای آن اجرا شده و اجرای نمونه به اتمام رسیده است [5].

$$L = \{\sigma_i : \sigma_i \in S, \quad 1 \leq i \leq K\} \quad (3)$$

که در آن S مجموعه‌ی تمام دنباله‌های ممکن و K تعداد دنباله‌های نگاره رویداد است.

پیشوند: یک دنباله مانند فرمول (2) تعریف شده است. اگر در این دنباله بخش ابتدایی آن در نظر گرفته شود، به آن پیشوند گفته می‌شود. به عبارتی دیگر تابع پیشوند با طول k در فرمول (4) نشان داده شده است [5]:

$$hd^k(\sigma) = \langle e_1, e_2, \dots, e_k \rangle, \quad k \leq n \quad (4)$$

پسوند: مشابه پیشوند، می‌توان پسوند را نیز مطابق فرمول (5) تعریف کرد. مجموع پسوند و پیشوند، کل دنباله را تشکیل می‌دهد [5].

$$tl^k(\sigma) = \langle e_{k+1}, e_{k+2}, \dots, e_n \rangle, \quad k \leq n \quad (5)$$

همچنین براساس فرمول (1) برای هر رویداد سه ویژگی شامل شناسه‌ی نمونه، فعالیت و برچسب زمانی اجباری است. بنابراین حداقل ویژگی‌هایی که یک رویداد را تعریف می‌کنند این است که چه نمونه‌ی اجرایی از فرایند (c) در چه زمانی (t)، چه فعالیتی (a) را انجام داده است [2]. با داشتن این سه ویژگی می‌توان تحلیل فرایندکاوی را آغاز کرد. اما در فرمول (1) سایر ویژگی‌ها نیز مطرح شده است که اگرچه اختیاری هستند، اما در صورت وجود می‌توانند تحلیل دقیق‌تر را ممکن سازند [5].

2-2- بیان مساله

برای تحلیل گلوگاه با استفاده از فرایندکاوی می‌توان از روش‌های مطرح شده در پشتیبانی عملیات بهره جست. از منظر پشتیبانی عملیات، هدف از تحلیل گاهی صرفا تشخیص است [4]. در مورد گلوگاه یعنی با در نظر گرفتن یک نمونه‌ی جاری از فرایند، مشخص شود قبلا در چه فعالیتی گلوگاه رخ داده است. سطح دیگر تحلیل، پیش‌بینی است. در این نوع تحلیل نگاه به آینده وجود دارد، بنابراین هدف تحلیل در این حالت پیش‌بینی گلوگاه در آینده است. نوع سوم تحلیل، پیشنهاد است. در این نوع تحلیل با در نظر گرفتن وضعیت جاری، مشخص می‌شود چه اقداماتی برای پیشگیری از رخ دادن گلوگاه در آینده لازم است انجام شود [4]. در این مقاله ما دومین نوع تحلیل (یعنی پیش‌بینی) را مدنظر قرار داده و به دنبال پیش‌بینی گلوگاه در ادامه‌ی اجرای نمونه‌ی فرایند هستیم.

برای پیش‌بینی گلوگاه باید به طور دقیق معنای گلوگاه را تعریف کنیم. در بخش قبل تعاریف متعدد گلوگاه مرور شد. در این پژوهش تعریفی از گلوگاه مدنظر قرار می‌گیرد که به دنبال یافتن فعالیت‌هایی در ادامه‌ی فرایند است که بیش از زمان مورد انتظار طول می‌کشند [4]. با بیانی دقیق‌تر گلوگاه فعالیت‌هایی هستند که فاصله‌ی مدت زمان انجام آن فعالیت و میانگین زمان آن فعالیت در تمامی نمونه‌ها نسبت به انحراف معیار، بیشتر از بقیه فعالیت‌ها در آن نمونه باشد. برای بیان ریاضی این تعریف، از رابطه (6) استفاده می‌کنیم. در این فرمول X، مدت زمان انجام یک فعالیت در نمونه‌ی جاری، μ میانگین مدت زمان اجرای این فعالیت در سایر نمونه‌ها و σ انحراف معیار است.

$$argmax\left(\frac{X-\mu}{\sigma}\right) \quad (6)$$

براساس فرمول (6)،یک نمونه‌ی جاری داریم که تا الان تعدادی رویداد از آن اجرا شده (پیشوندی به طول k) و تعدادی رویداد نیز باقی مانده است. هر رویداد بیانگر اجرای یک فعالیت مشخص است، در بین رویدادهای باقی‌مانده (پسوند)، به دنبال فعالیتی هستیم که اجرای آن، نسبت به همان فعالیت در سایر نمونه‌ها، مدت زمان بیشتری به طول خواهد انجامید. یعنی نمونه‌ی جاری، آن فعالیت خاص را کندتر از سایر نمونه‌ها در همان فعالیت خاص، اجرا خواهد کرد. حال بین همه‌ی فعالیت‌های آتی، فعالیتی گلوگاه است که $\frac{X-\mu}{\sigma}$ بیشتری نسبت به بقیه داشته باشد. یعنی فعالیتی در نمونه‌ی جاری که به طورغیرمعمول در مقایسه با سایر نمونه‌ها طول کشیده است.

2-3- پژوهش‌های پیشین

برای مرور پژوهش‌های پیشین، به دو بخش می‌پردازیم. نخست به پژوهش‌های انجام شده در تحلیل گلوگاه می‌پردازیم و سپس پژوهش‌هایی که به پیش‌بینی فعالیت بعدی و زمان باقی‌مانده می‌پردازند، مرور می‌کنیم.

از نظر نوع تحلیل گلوگاه مقالات را می‌توان به سه دسته تقسیم کرد [4]. دسته اول، روش‌های کشف گلوگاه هستند که در آنها، گلوگاهی که تاکنون رخ داده، کشف می‌شود [7,10]. دوم، روش‌های پیش‌بینی گلوگاه هستند که در آنها، برای یک نمونه‌ی جاری، پیش‌بینی می‌شود در ادامه چه گلوگاهی وجود خواهد داشت [13]. سوم، روش‌های پیشنهاد رفع گلوگاه هستند که در آنها، برای رفع گلوگاه راهکار و پیشنهاد ارائه می‌شود [14]. در این دسته‌بندی، تعداد کارهای انجام شده در حوزه‌ی کشف گلوگاه بیش از دو دسته‌ی دیگر است.

از جهت روش ارائه شده برای تحلیل گلوگاه نیز می‌توان مقالات را به سه دسته تقسیم کرد. اول، روش‌های مبتنی بر ابزارهای فرایندکاوی مانند دیسکو و پرام [11,12,16]. دوم، روش‌های مبتنی بر تحلیل‌های آماری [7,15] و سوم، روش‌های مبتنی بر یادگیری ماشین و یادگیری عمیق [13]. به عنوان مثال، در مقاله‌ی [13] پیش‌بینی گلوگاه با استفاده از یادگیری گروهی و با توجه به رانش مفهوم مورد توجه قرار گرفته است. رانش مفهوم به حالتی گفته می‌شود که رابطه بین داده ورودی و متغییر هدف دچار تغییر شود. در رویکرد ارائه شده توسط این مقاله، مجموعه داده به بازه‌های زمانی مختلف تقسیم شده و برای هر بازه‌ی زمانی یک مدل یادگیری ماشین آموزش داده می‌شود. در هنگام پیش‌بینی مدل‌های جدیدتر نسبت به مدل‌های قدیمی، وزن بیشتری می‌گیرند.

پایش پیش‌بینانه‌ی فرایند کسب و کار یکی از زیر شاخه‌های فرایندکاوی است که به پیش‌بینی درباره وضعیت آینده‌ی فرایند می‌پردازد [5]. وضعیت آینده‌ی فرایند می‌تواند انواع گوناگونی مانند پیش‌بینی زمان باقی مانده تا انتهای فرایند [5]، فعالیت بعدی یک نمونه‌ی اجرایی [17] و نتیجه‌ی نهایی [9] را شامل شود. در این پژوهش به دلیل کاربردی که در ادامه مدنظر است، تنها پژوهش‌هایی که علاوه بر فعالیت بعدی، زمان اجرا یا زمان باقی‌مانده را نیز پیش‌بینی کنند، مدنظر قرار می‌گیرند.

اخیرا اکثر روش‌های این حوزه، از یادگیری عمیق و مخصوصا شبکه‌های بازگشتی و حافظه‌ی طولانی کوتاه‌مدت (LSTM) استفاده می‌کنند [5]. همچنین یکی دیگر از معماری‌هایی که اخیرا مورد توجه قرار گرفته است، شبکه متخاصم مولد است [8]. در این روش، از دو شبکه‌ی مولد و تفکیک‌کننده استفاده می‌شود. شبکه ی مولد سعی در تولید داده هایی دارد که آنقدر شبیه به داده های واقعی باشند که وقتی این داده ها به شبکه ی تفکیک دهنده داده می شوند، نتواند داده های تولیدی را از داده های واقعی تشخیص دهد. هدف شبکه ی مولد به حداکثر رساندن دقت پیش بینی جهت فریب دادن تفکیک دهنده است. هدف شبکه ی تفکیک دهنده به حداقل رساندن خطا در تشخیص داده های واقعی از داده های تولید شده است.

در [8] به پیش‌بینی فعالیت‌های بعدی و زمان باقی‌مانده‌ی آن‌ها با استفاده از شبکه‌های متخاصم مولد که بهبود یافته‌ی [۴۱] است، پرداخته شده است. تفاوت این پژوهش با [۴۱] در این است که شبکه‌ی مولد دارای یک کدگذار و یک کدگشا از جنس شبکه‌های حافظه طولانی کوتاه‌مدت شامل لایه‌های کاملا متصل است. در نهایت برای پیش‌بینی گلوگاه از میان چندین گزینه، با استفاده از جستجوی پرتو گلوگاه انتخاب می‌شود.

3- روش پیشنهادی

3-1- شرح مساله، ورودی و خروجی

تعریف گلوگاه در این پژوهش، یافتن فعالیتی است که زمان اجرای آن به طور غیرمعمولی بیش از زمان مورد انتظار طول بکشد. برای پیش‌بینی گلوگاه می‌توان گلوگاه را به ازای یک فرایند (در کل نمونه‌ها) یا به ازای هر نمونه از فرایند پیش‌بینی کرد که در این مقاله، به ازای هر نمونه گلوگاه پیش‌بینی می‌شود. همچنین منظور از پیش‌بینی فعالیت، پیش‌بینی از بین فعالیت‌های باقی‌مانده است و فعالیت‌های قبلی در نظر گرفته نمی‌شوند.

به عنوان ورودی، نگاره رویداد دریافت شده و یک مدل از روی آن ساخته می‌شود که برای هر نمونه‌ی جاری می‌تواند مشخص می‌کند کدام فعالیت در آینده اتفاق افتاده و بیش از حد معمول طول میکشد.

3-2- ساختار کلی روش پیشنهادی

ساختار کلی روش پیشنهادی در شکل 1 نشان داده شده است. ابتدا به عنوان ورودی، نگاره رویداد در نظر گرفته شده و پیش‌پردازش‌هایی مانند کدگذاری و تجمیع برخی ویژگی‌ها روی آن انجام می‌شود. این پیش‌پردازش‌ها در ادامه با جزئیات بیشتری تشریح می‌شوند.

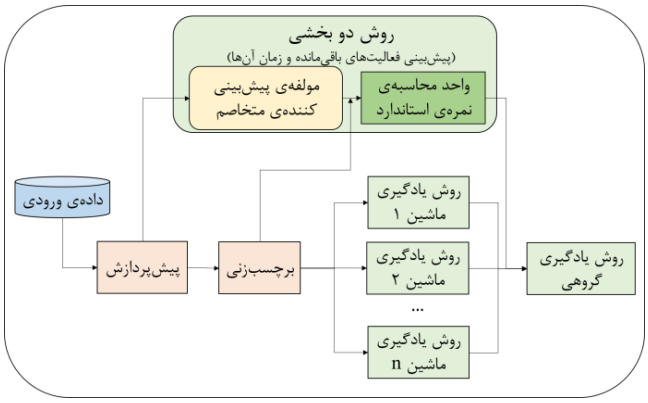
قدم بعد، برچسب‌زنی داده است. مجموعه داده به دو مجموعه‌ی آموزش و آزمون تقسیم شده و با استفاده از فرمول (6) و برای داده‌ی آموزش برچسب‌زنی انجام می‌شود. در واقع برای تمام فعالیت‌های پسوند در نمونه‌های آموزش مقدار نمره‌ی استاندارد با فرمول (7) محاسبه می‌شود:

$$z - score = \frac{X - \mu}{\sigma} \quad (7)$$

هر فعالیتی در پسوندِ نمونه‌ی موجود در داده‌های آموزشی که نمره‌ی استاندارد بیشتری در مقایسه با سایر فعالیت‌های پسوند داشته باشد، به عنوان گلوگاه آن نمونه انتخاب می‌شود.

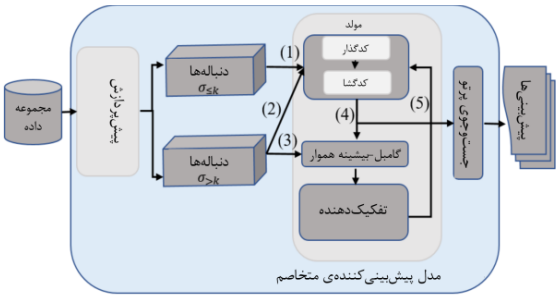
روش پیشنهادی این مقاله، یادگیری گروهی است. بنابراین تعدادی روش موجود است که هر کدام برای یک نمونه‌ی جاری، گلوگاهی را انتخاب می‌کنند و نهایتا این نتایج، توسط یادگیری گروهی با توجه به بیشترین رای تجمیع می‌شوند. همانطور که در شکل 1 مشاهده می‌شود، روش‌هایی که یادگیری گروهی از نتایج آن‌ها استفاده می‌کند، دو نوع هستند، یکی روش‌های موجود و متداول یادگیری ماشین شامل درخت تصمیم، جنگل تصادفی، XGBoost، رگرسیون منطقی و Kنزدیک‌ترین همسایه و روش دیگر، یک روش دو بخشی که در ادامه شرح داده شده و علت این نام‌گذاری نیز مشخص می‌شود.

در روش‌های یادگیری ماشین با دریافت داده‌های برچسب‌خورده، مدلی ایجاد می‌کنند که برای یک نمونه‌ی جاری، مستقیما پیش‌بینی می‌کند چه فعالیتی در آینده گلوگاه خواهد بود، اما در روش دو بخشی، نوع نگاه به مساله متفاوت است.



شکل 1 ساختار کلی روش پیشنهادی

ایده‌ی اصلی روش دو بخشی که توسط این مقاله به آن توجه شده، این است که مساله را به دو بخش تقسیم کنیم: در بخش اول، فعالیت‌های باقی‌مانده و زمان آن‌ها پیش‌بینی می‌شود و در بخش دوم با یک محاسبه‌ی آماری که در فرمول (6) شرح داده شد، مشخص کنیم کدام یک از این فعالیت‌های پیش‌بینی شده توسط بخش قبل گلوگاه خواهد بود. به بیانی دیگر، یکی از مسائل مهم حوزه‌ی فرایندکاوی، پایش پیش‌بینانه‌ی فرایندها است که به پیش‌بینی درباره وضعیت آتی فرایند شامل زمان باقی‌مانده، فعالیت(های) بعدی، نتیجه‌ی فرایند و غیره می‌پردازد. در این روش دو بخشی، در بخش اول از مساله‌ی پیش‌بینی فعالیت‌های باقی‌مانده و زمان آن‌ها، استفاده می‌شود، در نتیجه‌ی این پیش‌بینی، بخش دوم صرفا به محاسبه‌ی نمره‌ی استاندارد می‌پردازد و از بین فعالیت‌های پیش‌بینی شده، فعالیتی را انتخاب می‌کند که به طور غیرمعمول پیش‌بینی می‌شود، بیشتر طول می‌کشد. در ادامه، روش دو بخشی با جزئیات بیشتری تشریح می‌شود.



شکل 2 معماری مولفه‌ی پیش‌بینی کننده‌ی متخاصم مولد [8]

همانطور که در شکل 1 مشاهده می‌شود، در بخش اول روش دو بخشی، مولفه‌ی پیش‌بینی کننده‌ی متخاصم مولد وجود دارد. داده‌ی پیش‌پردازش شده به یک شبکه‌ی متخاصم مولد داده می‌شود. این شبکه فعالیت‌های بعدی و زمان اجرای هر کدام را برای یک نمونه‌ی جاری پیش‌بینی می‌کند.

معماری مولفه‌ی پیش‌بینی کننده‌ی متخاصم مولد در شکل 2 ترسیم شده است. دو واحد ساخت دنباله وجود دارد. یک واحد برای ساخت دنباله‌ی پیشوندها به طول کمتر و مساوی k و واحد دیگر برای ساخت دنباله‌ی پسوندها به طول بیشتر از k است. پس از ساخت پیشوندها و پسوندها، داده‌ها به مدل‌های مولد و تفکیک‌دهنده داده می‌شوند. مدل مولد با استفاده از کدگذار و کدگشا که از نوع شبکه‌ی حافظه طولانی کوتاه مدت (LSTM) است، پیش‌بینی‌هایی انجام می‌دهد. سپس این پیش‌بینی‌ها به تابع گامبل-بیشینه هموار داده می‌شوند. این تابع، پیش‌بینی‌ها را به صورت یک داده‌ی کامل درمی‌آورد؛ یعنی پیشوند، پسوند و زمان را کنار هم قرار داده و به مدل تفکیک‌کننده می‌دهد.

مدل تفکیک‌دهنده باید بتواند این پیش‌بینی‌ها را از داده‌ی واقعی تشخیص دهد. اگر نتواند، پس مولد به آن حد از پیش‌بینی دست یافته که می‌تواند تفکیک‌دهنده را فریب دهد. در نهایت پیش‌بینی‌ها به واحد جستجوی پرتو داده می‌شوند. در جستجوی پرتو از میان n فعالیت پیش‌بینی شده و با محاسبه‌ی امتیاز، فعالیتی که بیشترین امتیاز را داشته باشد، انتخاب می‌شود [8]. در نهایت مدل پیش‌بینی در خروجی به واحد محاسبه نمره استاندارد تحویل داده می‌شوند. این واحد روی داده‌ی آزمون مدل پیش‌بینی را اجرا می‌کند و به ازای هر نمونه‌ی جاری، پسوند دنباله را پیش‌بینی می‌کند، به عبارتی دیگر چه فعالیت‌هایی در ادامه رخ خواهند داد و زمان اجرای آن‌ها چقدر خواهد بود. سپس با داشتن این اطلاعات نمره‌ی استاندارد را برای هر کدام از فعالیت‌های پسوند، محاسبه کرده و فعالیتی که نمره‌ی استاندارد بالاتری داشته باشد به آن معناست که بیش از حد معمول طول کشیده بنابراین به عنوان گلوگاه معرفی می‌شود.

سرانجام پس از آن که نتایج گلوگاه توسط روش‌های یادگیری ماشین و روش دو بخشی پیدا شد، یادگیری گروهی بین این نتایج، رای‌گیری می‌کند. برای پیاده‌سازی یادگیری گروهی، تمام ترکیب‌های 4 الی 6 تایی روش‌های یادگیری ماشین و روش دو بخشی بررسی شده و نتیجه‌ی بهترین ترکیب ممکن، جنگل تصادفی، XGBoost، رگرسیون منطقی و روش دو بخشی است؛ همچنین در صورت برابری رای‌ها نیز اولویت با روش دو بخشی است.

3-3- جزئیات پیاده‌سازی

در پیاده‌سازی‌های انجام شده دو نوع پیش‌پردازش وجود دارد: پیش‌پردازش مربوط به شبکه‌ی متخاصم مولد و پیش‌پردازش مربوط به واحد محاسبه‌ی نمره‌ی استاندارد. برای شبکه‌ی متخاصم مولد، فقط از سه ویژگی اجباری نگاره رویداد شامل شناسه نمونه، فعالیت و برچسب زمانی و برای کدگذاری فعالیت نیز از one-hot استفاده شده است. همچنین محاسباتی برای مدت زمان اجرا و زمان باقی‌مانده نیز انجام شده است. برای پیش‌پردازش واحد محاسبه‌ی نمره‌ی استاندارد نیز علاوه بر پیش‌پردازش‌هایی مانند کدگذاری، با توجه به طول پیشوند که در ادامه تشریح می‌شود، پیشوندهای مختلف تولید می‌شوند.

در روش‌های فرایندکاوی، ورودی یک دنباله‌ی ناتمام از رویدادهای یک نمونه است، این در حالیست که نگاره رویداد شامل اجرای کامل تمام نمونه‌هاست. بنابراین برای آموزش و آزمون، باید هر بار پیشوندی از هر نمونه را در نظر گرفته و براساس آن مدل را آموزش داده و سپس بسنجیم. در اغلب پیاده‌سازی‌ها طول پیشوند یک عدد ثابت در نظر گرفته نمی‌شود بلکه یک حد پایین و بالا برای آن تعیین می‌شود و تمام حالت‌های طول پیشوند محاسبه می‌شوند. در پیاده‌سازی این مقاله حد پایین طول پیشوندها برابر 2 و برای حد بالا نیز کمینه‌ی بین دو مقدار در نظر گرفته می‌شود: اول، تعداد فعالیت‌ها منهای یک و دوم، دهک نهم در مجموعه‌ی طول فعالیت‌های هر نمونه.

4- مجموعه داده و ارزیابی

4-1- مجموعه داده

مجموعه داده «میزکمک» شامل نمونه‌های اجرایی از فرایند مدیریت درخواست‌های پشتیبانی یک شرکت نرم‌افزاری در ایتالیا است. همه‌ی نمونه‌ها با فعالیت «ثبت درخواست جدید» آغاز شده و با یکی از فعالیت‌های «حل شدن مشکل» یا «بسته شدن درخواست» به اتمام می‌رسند. بازه زمانی این نگاره رویداد، از سال 2010 تا 2014 است و شامل 14 فعالیت، 4580 نمونه و 21348 رویداد است.

4-2- معیارهای ارزیابی

فرض کنیم TP تعداد تشخیص‌های مثبت درست، TN تعداد تشخیص‌های منفی درست، FP تعداد تشخیص‌های مثبت اشتباه و FN تعداد تشخیص‌های منفی اشتباه است، در این صورت صحت برابر نسبت تعداد تشخیص‌های درست به تمام تشخیص‌ها است و در فرمول (8) بیان شده است.

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (8)$$

زودکرد در حل مسائل فرایندکاوی از نوع پیش‌بینی موردتوجه قرار می‌گیرد. مفهوم زودکرد این است که هر چه پیش‌بینی در پیشوندهای با طول کمتر بهتر انجام شود، ارزشمندتر است. بنابراین به نمونه‌های با طول پیشوند کمتر وزن بیشتری نسبت داده می‌شود [5,9]. تعیین وزن‌ها به این صورت است که برای مثال پیشوندهای به طول ۲ وزنی به طول آخرین پیشوند که ۶ است، می‌گیرند و پیشوندهای به طول ۶ وزن کوتاه‌ترین پیشوند یعنی ۲ را می‌گیرند و برای بقیه‌ی پیشوندها نیز به همین روال محاسبه انجام می‌شود. در نهایت یک میانگین وزن‌دار از معیار ارزیابی گرفته می‌شود [9].

4-3- روش‌های پایه

در بخش ارزیابی، روش‌های یادگیری ماشین و یک روش پایه برای یادگیری گروهی به عنوان روش‌های پایه در نظر گرفته شده‌اند. روش‌های یادگیری ماشین شامل جنگل تصادفی، XGBoost، رگرسیون منطقی، Kنزدیک‌ترین همسایه، درخت تصمیم و روش یادگیری گروهی پایه شامل ترکیب روش‌های XGBoost، جنگل تصادفی و رگرسیون منطقی در نظر گرفته شده است. برخی از پارامترهای این روش‌ها که با استفاده از کتابخانه‌ی scikit-learn پایتون پیاده‌سازی شده‌اند، در جدول 1 مشاهده می‌شود.

جدول 1 مقادیر برخی پارامترهای مدل‌های یادگیری ماشین

مدل	پارامتر	مقدار	توضیح
جنگل تصادفی	n_estimators	200	تعداد درخت‌ها در جنگل
	criterion	entropy	تابع سنجش کیفیت تقسیم درخت
XGBoost	objective	softmax multi	تابع هدف
	max_depth	6	حداکثر عمق تخمین‌گرها
	learning_rate	0.3	نرخ یادگیری برای تعیین مشارکت هر درخت
	n_estimators	200	تعداد مراحل انجام boosting
رگرسیون منطقی	C	2	پارامتر regularization
	max_iter	300	حداکثر تعداد تکرار
K نزدیک‌ترین همسایه	n_neighbors	5	تعداد همسایه‌ها
	metric	minkoeski	معیار سنجش فاصله
درخت تصمیم	criterion	gini	تابع سنجش کیفیت تقسیم درخت

4-4- نتایج ارزیابی

در جدول 2 نتایج ارزیابی صحت در روش‌های پایه با در نظر گرفتن زودکرد و بدون آن مشاهده می‌شود. با توجه به این جدول، بالاترین صحت متعلق به روش یادگیری گروهی پایه (شامل XGBoost، جنگل تصادفی و رگرسیون منطقی) است.

جدول 2 ارزیابی صحت روش‌های پایه

مدل یادگیری ماشین	درصد صحت (با در نظر گرفتن زودکرد)	درصد صحت (بدون در نظر گرفتن زودکرد)
درخت تصمیم	51.75	50.82
رگرسیون منطقی	32.21	48.63
جنگل تصادفی	58.67	52.46
XGBoost	62.72	55.05
Kنزدیک‌ترین همسایه	41.23	39.89
یادگیری گروهی پایه	**63.96**	**57.10**

در جدول 3 نتایج ارزیابی روش دو بخشی که توسط این مقاله برای حل مساله پیش‌بینی گلوگاه استفاده شده و همچنین روش یادگیری گروهی پیشنهادی مشاهده می‌شود. برای یافتن بهترین ترکیب روش‌ها در روش یادگیری گروهی، آزمایش‌هایی انجام و نهایتا مشخص شد ترکیب روش دو بخشی، جنگل تصادفی، XGBoost و رگرسیون منطقی بهترین ترکیب است. به دلیل اینکه ترکیب 4 تایی است، در صورت برابری رای‌ها، به روش دو بخشی اولویت بالاتر داده شد. همانطور که در جدول 3 مشاهده می‌شود، در چنین شرایطی روش پیشنهادی با اختلاف قابل‌توجهی می‌تواند بهتر از سایر روش‌ها عمل کند. همچنین روش دو بخشی عملکردی مشابه یادگیری گروهی پایه دارد و البته صحت آن بدون در نظر گرفتن زودکرد بهتر از یادگیری گروهی پایه است. نتایج کلیه‌ی روش‌ها در نمودار شکل 3 ترسیم شده است.

جدول 3 ارزیابی صحت روش دو بخشی و روش پیشنهادی

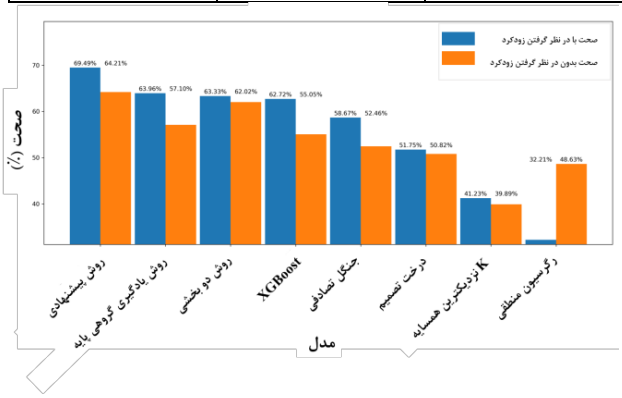
مدل یادگیری ماشین	درصد صحت (با در نظر گرفتن زودکرد)	درصد صحت (بدون در نظر گرفتن زودکرد)
روش دو بخشی	63.32	62.02
روش پیشنهادی	**64.21**	**69.49**

جهت اطمینان از معنادار بودن اختلاف روش پیشنهادی با سایر روش‌ها آزمون مک‌نمار انجام شده است. آزمون آماری مک‌نمار یک آزمون ناپارامتری است که برای تحلیل دادگان عددی دوپاسخی به کار می‌رود. در پژوهش‌های فرایندکاوی، این آزمون بسیار پرکاربرد است [5,9]. در این آزمون، ابتدا ماتریس درهم‌ریختگی از جواب‌های دو نمونه ساخته می‌شود، سپس طبق فرمول (9) مقدار p محاسبه می شود که اگر دو نمونه همانند نباشند باید این مقدار کمتر از 0.05 باشد. در این فرمول D برابر FN و A برابر FP است [42]. همانطور که در جدول 4 مشاهده می‌شود، اختلاف روش پیشنهادی با چهار روشی که جزئی از آن هستند (از جمله روش دو بخشی) و همچنین روش یادگیری گروهی پایه که بهترین عملکرد در میان روش‌های پایه دارد، معنادار است.

$$\frac{(D-A)^2}{D+A} \qquad (9)$$

جدول 4 آزمون مک‌نمار برای ارزیابی معنادار بودن نتایج

مدل اول	مدل دوم	مقدار p
روش پیشنهادی	روش دو بخشی	0.0375
روش پیشنهادی	جنگل تصادفی	2.2234e-09
روش پیشنهادی	XGBoost	4.9874e-06
روش پیشنهادی	رگرسیون منطقی	7.9261e-23
روش پیشنهادی	روش یادگیری گروهی پایه	0.04114



شکل 3 مقایسه‌ی صحت در روش‌های مختلف

5- نتیجه‌گیری و کارهای آتی

در این مقاله به حل مساله‌ی تحلیل گلوگاه‌ها با استفاده از روش‌های فرایندکاوی پرداختیم. ابتدا تعاریف متعدد گلوگاه بررسی شده و نهایتا از تعریفی استفاده شد که به زمان توجه ویژه دارد. بنابراین منظور از گلوگاه فعالیت‌هایی هستند که بیش از زمان مورد انتظار طول بکشند. علاوه بر تعریف گلوگاه، موضوع مهم بعدی تعیین نوع تحلیل است که در این مقاله، تحلیل از نوع پیش‌بینی مدنظر قرار گرفت. اکثر پژوهش‌های پیشین از روش‌های آماری

برای پیش‌بینی گلوگاه استفاده کرده‌اند و سهم خیلی کمتری به روش‌های یادگیری ماشین اختصاص دارد اما در این مقاله روش‌های یادگیری ماشین و یادگیری عمیق کاملا مورد توجه قرار گرفته‌اند. در گام بعد تعدادی روش یادگیری ماشین به عنوان روش پایه و یک روش یادگیری گروهی پایه پیاده‌سازی شد. همچنین در این مقاله، تلاش کردیم مساله‌ی پیش‌بینی گلوگاه را به کمک تبدیل به مساله‌ی پیش‌بینی فعالیت‌ها و زمان آن‌ها نیز حل کنیم، در نتیجه روش دوبخشی را معرفی کردیم. سرانجام یک روش یادگیری گروهی پیشنهاد شد که ترکیبی از روش دو بخشی و سایر روش‌های یادگیری ماشین است و نشان داده شد که این روش عملکرد بهتری در مقایسه با سایر روش‌های پایه دارد و جهت اطمینان از نتایج نیز آزمون آماری انجام شد که معنادار بودن نتایج بهبود را تایید کرد.

افزودن مدل‌های دیگر به روش یادگیری گروهی و ارزیابی تاثیر آن‌ها، بررسی روش پیشنهادی روی سایر مجموعه داده‌ها، تغییر در فرمول محاسبه‌ی نمره‌ی استاندارد و روش برچسب‌زنی از جمله مواردی است که در ادامه‌ی این کار پژوهشی در نظر گرفته خواهند شد.

مراجع

[1] Dumas, Marlon, Marcello La Rosa, Jan Mendling, and Hajo A. Reijers. *Fundamentals of business process management*. Vol. 2. Heidelberg: Springer, 2018.

[2] Van Der Aalst, Wil, and Wil van der Aalst. *Data science in action.* Springer Berlin Heidelberg, 2016.

[3] Van Der Aalst, Wil MP. *"Process discovery from event data: Relating models and logs through abstractions."* Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 8, no. 3 (2018): e1244.

[4] Bemthuis, Rob H., Niels van Slooten, Jeewanie Jayasinghe Arachchige, Jean Paul Sebastian Piest, and Faiza Allah Bukhsh. *"A Classification of Process Mining Bottleneck Analysis Techniques for Operational Support."* In ICE-B, pp. 127-135. 2021.

[5] Verenich, Ilya, Marlon Dumas, Marcello La Rosa, Fabrizio Maria Maggi, and Irene Teinemaa. *"Survey and cross-benchmark comparison of remaining time prediction methods in business process monitoring." ACM Transactions on Intelligent Systems and Technology (TIST)* 10, no. 4 (2019): 1-34.

[6] Roser, Christoph, Kai Lorentzen, and Jochen Deuse. *"Reliable shop floor bottleneck detection for flow lines through process and inventory observations: the bottleneck walk."* Logistics Research 8 (2015): 1-9.

[7] Heo, Gwangjin, Jinsung Lee, and Jae-Yoon Jung. *"Analyzing bottleneck resource pools of operational process using process mining."* ICIC express letters. Part B, Applications: an international journal of research and surveys 9, no. 5 (2018): 437-441.

[8] Taymouri, Farbod, Marcello La Rosa, and Sarah M. Erfani. *"A deep adversarial model for suffix and remaining time prediction of event sequences."* In Proceedings of the 2021 SIAM International Conference on Data Mining (SDM), pp. 522-530. Society for Industrial and Applied Mathematics, 2021.

[9] Teinemaa, Irene, Marlon Dumas, Marcello La Rosa, and Fabrizio Maria Maggi. *"Outcome-oriented predictive process monitoring: Review and benchmark."* ACM Transactions on Knowledge Discovery from Data (TKDD) 13, no. 2 (2019): 1-57.

[10] Kumbhar, Mahesh, Amos HC Ng, and Sunith Bandaru. *"Bottleneck detection through data integration, process mining and factory physics-based analytics."* In 10th Swedish Production Symposium (SPS2022), Skövde, April 26–29 2022, pp. 737-748. IOS Press, 2022.

[11] Rashed, Abdel-Hamed Mohamed, Noha E. El-Attar, Diaa Salama Abdelminaam, and Mohamed Abdelfatah. *"Analysis the patients' careflows using process mining."* Plos one 18, no. 2 (2023): e0281836.

[12] Rudnitckaia, Mgr JULIA., *Modelling and Analysis of Logistics Processes by applying Process and Data Mining Techniques*, Ph.D. Thesis, Brno University of Technology, Brno, Czech Republic, 2022.

[13] Spenrath, Yorick, and Marwan Hassani. *"Ensemble-Based Prediction of Business Processes Bottlenecks With Recurrent Concept Drifts."* In EDBT/ICDT Workshops. 2019.

[14] Ahmed, Razi, Muhammad Faizan, and Anwer Irshad Burney. *"Process mining in data science: A literature review."* In 2019 13th International Conference on Mathematics, Actuarial Science, Computer Science and Statistics (MACS), pp. 1-9. IEEE, 2019.

[15] AlBakary, Wessam Ahmed, Ahmed Ahmed Hesham Sedky, and Walid Abdelmoez. *"Interactive Bottleneck Detection in Data Driven Business Process Simulation in Healthcare: Egyptian Case Study."* In 2022 32nd International Conference on Computer Theory and Applications (ICCTA), pp. 234-240. IEEE, 2022.

[16] Caballero-Hernández, Juan Antonio, Juan Manuel Dodero, Iván Ruiz-Rube, Manuel Palomo-Duarte, José Fidel Argudo, and Juan José Domínguez-Jiménez. "Discovering bottlenecks in a computer science degree through process mining techniques." In 2018 International Symposium on Computers in Education (SIIE), pp. 1-6. IEEE, 2018.

[17] Tax, Niek, Ilya Verenich, Marcello La Rosa, and Marlon Dumas. *"Predictive business process monitoring with LSTM neural networks."* In Advanced Information Systems Engineering: 29th International Conference, CAiSE 2017, Essen, Germany, June 12-16, 2017, Proceedings 29, pp. 477-492. Springer International Publishing, 2017.

پنجمین کنفرانس ملی انفورماتیک ایران
پژوهشگاه دانشهای بنیادی، پردیس فرمانیه، تهران
۱۳ و ۱۴ دی ماه ۱۴۰۲

ارائه طرح تشویقی و اولویت‌بندی وظایف جهت استفاده بهینه از انرژی مازاد خودروهای الکتریکی در محاسبات مه به کمک کنترل‌کننده SDN

فائزه رحمانی[۱]، نیک محمد بلوچزهی[۲]

[۱] کارشناسی ارشد، دانشکده مهندسی برق و کامپیوتر، دانشگاه سیستان و بلوچستان، زاهدان،
rahmani.faeze@gmail.com

[۲] استادیار، دانشکده مهندسی برق و کامپیوتر، دانشگاه سیستان و بلوچستان، زاهدان
balouchzahi@ece.usb.ac.ir

چکیده

شبکه‌های خودرویی یکی از مولفه‌های اصلی سیستم‌های حمل و نقل هوشمند با کاربردهایی در حوزه‌های ایمنی، ترافیکی و رفاهی کمک شایانی در به اشتراک‌گذاری و توزیع اطلاعات در محیط شهری می‌کنند. از طرفی با پیشرفت صنعت خودروهای الکتریکی و افزایش توان محاسباتی و ذخیره‌سازی، این خودروها قادر خواهند بود علاوه‌بر پشتیبانی از خدمات سرگرمی و مدیریت ایمنی، یک مدل سودمند از محاسبات یعنی پردازش و ذخیره‌سازی را به کاربران ارائه دهند اما این خودروها زمان زیادی را در حالت پارک به سر می‌برند. در نتیجه می‌توان از انرژی مازاد آن‌ها برای انجام کارهای محاسباتی بهره برد. عدم وجود انگیزه‌های کارآمد و راه‌کارهای تعیین تکلیف کار با توجه به تنوع درخواست‌ها در شبکه از جمله چالش‌های موجود در این زمینه است.

در این پژوهش یک رویکرد برای اولویت‌بندی درخواست‌ها با توجه به درنظر گرفتن مهلت زمانی آن‌ها ارائه می‌گردد. هم‌چنین با اجرای یک طرح تشویقی برای خودروهایی که منابع در اختیار شبکه می‌گذارند باعث افزایش میزان انرژی در شبکه می‌شود. ارزیابی و آزمایشات صورت گرفته با ابزارهای شبیه‌سازی نشان‌دهنده بهبود ۱۵.۶۹ درصدی نرخ پاسخ موفق، ۱۱.۰۲ درصدی بهره‌وری منابع و کاهش۱.۱۴ ثانیه‌ای میانگین زمان تاخیر درخواست‌ها در روش پیشنهادی نسبت به راه‌کار پایه است.

کلمات کلیدی

خودروهای الکتریکی، تخصیص منابع، شبکه‌های نرم‌افزار محور، مدل تشویقی، راه‌کار زمان‌بندی کارها.

۱- مقدمه

تخمین زده شده است که تعداد دستگاه‌های هوشمند مستقر متصل از طریق اینترنت اشیا[۱] تا سال ۲۰۲۵ حدود ۷۵ میلیارد خواهد بود[1]. در حالی که انتظار می‌رود میزان داده تولید شده توسط این دستگاه‌ها ۷۳.۱ زتابایت در سال باشد[2, 3]. بیشتر این حجم عظیم از داده‌ها به پردازش بلادرنگ برای تصمیم‌گیری کارآمد نیاز دارند که پهنای باند بی‌سیم فعلی مورد استفاده در این شبکه‌ها جوابگوی تبادل اطلاعات حجیم نخواهد بود و شبکه اشباع خواهد شد[4]. پیش از این، از رایانش ابری به‌عنوان روشی کارآمد برای ذخیره سازی و انجام محاسبات بر روی حجم عظیمی از داده‌های تولید شده استفاده می‌شد. اما امروزه، به‌دلیل محدودیت‌های پهنای باند، تأخیر بالا و دور بودن از محل درخواست خدمت، قادر به پاسخگویی موثر به میلیاردها درخواست دستگاه‌های هوشمند نیست[5]. بنابراین، برای حل محدودیت‌های محاسبات ابری، رویکرد جدیدی به نام محاسبات مه[۲] ارائه شد که در آن داده‌ها، پردازش و خدمات، در لبه شبکه متمرکز شده‌اند. با این حال، خدمات محاسبات مه به‌دلیل تراکم واحدهای کنار جاده‌ای محدود است و واحدهای کنار جاده‌ای با افزایش درخواست خدمات، با بار سنگین روبرو می‌شوند[6].

با پیشرفت سریع تکنولوژی، واحدهای OBU[۳] خودروهای الکتریکی را قادر می‌سازد که علاوه بر پشتیبانی از سرگرمی و مدیریت ایمنی خودرو، یک مدل سودمند از محاسبات یعنی پردازش و ذخیره‌سازی را به کاربران شبکه ارائه دهند[7]. تعداد خودروهای برقی در خیابان‌ها در حال افزایش است. با این حال، این خودروها ۹۰ درصد از زمان را پارک هستند و منابع آن‌ها بدون استفاده است[5, 7]. به‌منظور یکپارچه‌سازی شبکه محاسبات مه و خودروها، محاسبات مه خودرویی نویدبخش دستیابی به پاسخ‌های شبکه مناسب‌تر در زمان واقعی و آگاه از مکان است[8]. در نتیجه، می‌توان خودروهای الکتریکی پارک شده را به‌عنوان سرورهای محاسباتی(گره‌های مه) درنظرگرفت و از منابع کم مصرف آن‌ها برای خدمات محاسباتی، استفاده کرد و کارهای محاسباتی را از ایستگاه پایه به گره‌های مه تخلیه کرد که این راه‌حلی برای کاهش بار ایستگاه‌های پایه وکاهش تاخیر درشبکه است.

معماری فعلی نیازهای مختلفی مانند انعطاف‌پذیری و مقیاس‌پذیری را که مورد نیاز برنامه‌های سیستم حمل‌ونقل هوشمند است، برآورده نمی‌کند. در نتیجه مشکلاتی در زمینه استقرار و مدیریت سیستم فعلی وجود دارد. از طرفی، شبکه نرم‌افزارمحور[۴](SDN) چشم‌انداز جدیدی در مورد نحوه مدیریت شبکه است که می‌تواند به کمک مولفه‌های نرم‌افزاری متمرکز، رفتار شبکه را به‌صورت پویا تغییر، مدیریت و کنترل کند[9]. ایده اصلی این روش، جدا کردن سخت‌افزار از نرم‌افزار است. بنابراین می‌تواند وظایف کنترل و ارسال بسته‌ها را به‌صورت جداگانه انجام دهد. این معماری، تأثیر قطع ارتباط ناشی از

تحرک خودروها را کاهش می‌دهد و قابلیت اطمینان ارتباطات شبکه‌های خودرویی را افزایش می‌دهد[9].

در اکثر مطالعات گذشته فرض شده است که خودروها منابع محاسباتی خود را بدون قید و شرط به اشتراک می‌گذارند[10]، که این فرض در واقعیت بسیار خوشبینانه است. با توجه به هزینه‌های لازم برای پردازش کارها، خودروها در صورتی علاقه‌مند هستند که نقش گره مه را ایفا کنند که این هزینه‌ها به خوبی جبران شوند. برخی از این طرح‌های تشویقی در جدول (۱) آورده شده است. برای استفاده از منابع خودروهای پارک شده، فرض شده صاحبان خودروها منابع را رایگان در اختیار شبکه می‌گذارند یا پاداش‌هایی نظیر پرداخت پولی، کسر از هزینه پارکینگ و غیره درنظر گرفته شده که انگیزه کافی نداشتند. بنابراین نیاز به راه‌کار تشویقی می‌باشد که علاوه‌بر افزایش انگیزه، میزان انرژی در شبکه را نیز افزایش دهد.

جدول (۱) : طرح‌های تشویقی ارائه شده

مرجع	سال	نوع تشویق ارائه شده
[6]	۲۰۱۹	پرداخت پول
[10]	۲۰۱۹	پرداخت پاداش توسط ایستگاه پایه با توجه به میزان منابع ارائه شده توسط خودروها
[11]	۲۰۲۰	کاهش هزینه پارک و پارک رایگان
[12]	۲۰۲۰	پرداخت پول
[13]	۲۰۲۰	پرداخت پول
[14]	۲۰۲۰	تنظیم قرارداد با بهینه‌سازی قیمت برای اپراتور سرویس داده و خودروها

مطالعه کارهای گذشته (جدول۲) و هم‌چنین نتایج موجود در مطالعه [15]که حاصل بررسی صد راه‌کار زمان‌بندی است نشان می‌دهد که تاخیر، هزینه و زمان اجرا مهم‌ترین معیارهای ارزیابی در بیشتر الگوریتم‌های زمان‌بندی هستند که هر کدام به ترتیب حدود ۱۷٪ ، ۱۲٪ و ۱۱٪ را شامل می‌شوند. از طرف دیگر، برنامه‌ها از لحاظ نوع و کیفیت خدمات متفاوت هستند. بنابراین مناسب است که یک راه‌کار زمان‌بندی کار طراحی شود که از اولویت برنامه‌ها استفاده کند. درخواست‌ها با اولویت کمتر منتظر می‌مانند تا کار با اولویت بالاتر تکمیل شود یا به ابر هدایت شود. با این وجود، وظایف با اولویت بالاتر در بدو ورود، انجام می‌شوند و این باعث کاهش تاخیر و بهبود کیفیت خدمات می‌شود. بنابراین مناسب است که یک راه‌کار زمان‌بندی کار طراحی شود که از اولویت برنامه‌ها استفاده کند و کارها با مهلت زمانی کم را در اولویت قرار دهد.

جدول (۲) : مرور راه‌کارهای زمان‌بندی

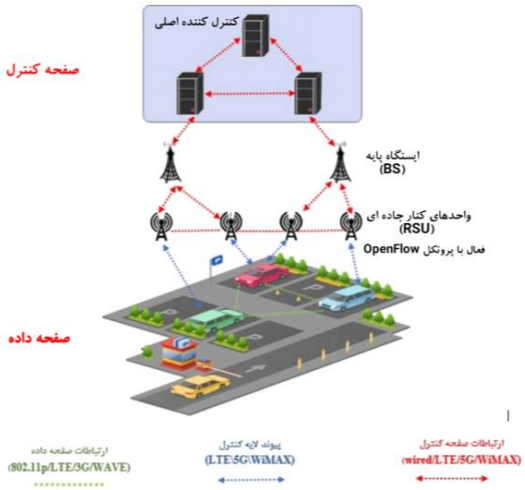
مرجع	سال	پارامتر موردنظر
[16]	۲۰۱۷	تحرک، ظرفیت
[17]	۲۰۱۸	کیفیت خدمات، تاخیر
[18]	۲۰۱۹	مهلت کارها
[19]	۲۰۲۰	مهلت کارها
[13]	۲۰۲۰	تاخیر

از آن‌جایی که ارائه انواع خدمات در محیط شهری در کمترین زمان ممکن و با تاخیرکم، نیازمند توان پردازشی بالا می‌باشد در نتیجه می‌توان از بستر شبکه‌های خودرویی برای رسیدن به این هدف کمک گرفت. با توجه به اینکه منابع خودروهای الکتریکی محدودانـد و درخواست‌ها از لحاظ کیفیت خدمات متفاوت هستند وکاربران علاقه‌مندند که درخواست‌هایشان درکمترین زمان ممکن پاسخ داده شود، پس می‌توان با ارائه یک راه‌کار زمان‌بندی، کارها را اولویت‌بندی کرد که این امر باعث کاهش تاخیر در کارها با اولویت بالا می‌شود. یکی دیگر از چالش‌های موجود در شبکه‌های خودرویی، عدم توجه کافی به راه‌کارهای تشویقی است. از طرفی روش‌های ارائه شده در محیط‌های شهری و بزرگ از مقیاس‌پذیری کمی برخودار هستند و دید کلی و دقیق از وضعیت شبکه ندارند. در این مقاله با ارائه راه‌کاری مبتی‌بر شبکه‌های نرم‌افزارمحور مقیاس‌پذیری را افزایش داده‌است. کنترل‌کننده شبکه‌های نرم‌افزار محور در هر لحظه یک دید جامع از درخواست‌ها و به‌طور کلی گراف شبکه دارد که با کنترل تمامی رفتارهای شبکه در کمترین زمان ممکن می‌تواند به درخواست‌ها پاسخ دهد

۲- روش پیشنهادی

پیشرفت فناوری در صنعت باتری خودروهای الکتریکی در سال‌های اخیر، باعث افزایش محبوبیت آن‌ها شده‌است و در نتیجه باعث عرضه خودروهای الکتریکی با ظرفیت محاسباتی و ذخیره سازی بالا می‌شود. بنابراین، تعداد خودروهای برقی در خیابان‌ها در حال افزایش است. با این حال، این خودروها ۹۰ درصد از زمان را پارک هستند و منابع آن‌ها بدون استفاده است[5]. از آن‌جایی‌که ارائه انواع خدمات در محیط شهری در کمترین زمان ممکن و با تاخیرکم، نیازمند توان پردازشی بالا می‌باشد. در نتیجه، استفاده از توان باتری اضافیِ خودروهای الکتریکیِ پارک شده در محیط شهری به منظور انجام پردازش‌های موردنیاز جهت ارائه انواع سرویس‌ها یکی از راه‌های پاسخگویی به این نیاز می‌باشد. معماری فعلی نیازهای مختلفی مانند انعطاف پذیری و مقیاس پذیری را که مورد نیاز برنامه‌های سیستم حمل‌ونقل هوشمند است، برآورده نمی‌کند در نتیجه مشکلاتی در زمینه استقرار و مدیریت سیستم فعلی وجود دارد.

در روش پیشنهادی از شبکه نرم‌افزار محور خودرویی استفاده شده است. مهم‌ترین ویژگی این شبکه جدا بودن صفحه کنترل از صفحه داده است که امکان کنترل و مدیریت سخت‌افزار را از طریق یک نرم‌افزار متمرکز دارد. یک کنترل‌کننده منطقی متمرکز وجود دارد که نمای گسترده ای از شبکه دارد و چندین دستگاه ارسال بسته (به عنوان مثال سوئیچ‌ها) را کنترل می‌کند که می‌توانند از طریق یک رابط مانند OpenFlow پیکربندی شوند. معماری ترکیب شده شبکه خودرویی با SDN قطع ارتباط ناشی از تحرک خودروها را کاهش می‌دهد و قابلیت اطمینان ارتباطات در شبکه خودرویی را افزایش می‌دهد. در این روش خودروهای موجود در پارکینگ توسط نقاط دسترسی موجود در محدوده خود به شبکه متصل می‌شوند. این نقاط دسترسی، سوییچ‌های مجهز به پروتکل OpenFlow هستند که از طریق این پروتکل با کنترل‌کننده شبکه ارتباط برقرار می‌کنند. در شکل (۱) معماری روش پیشنهادی نشان داده شده است.



شکل (۱) : معماری روش پیشنهادی

خودروهای دارای منابع مازاد، میزان انرژی که خودرو می‌تواند در اختیار شبکه بگذارد، مدت زمان پارک (مدت زمان تقریبی ایستادن خودرو در پارکینگ)، توان پردازشی پردازنده برحسب تعداد دستوراتی که در واحد زمان (MIPS)[۵] می‌تواند انجام دهد و همچنین میزان انرژی مصرفی به ازای هر دستورالعمل بر حسب کیلووات ساعت را به کنترل‌کننده اعلام می‌کنند. کنترل‌کننده در هرلحظه ظرفیت پارکینگ‌ها را بررسی می‌کند و هر زمان که خودروی جدیدی برسد، ظرفیت پارکینگ‌ها را به خودرو اعلام می‌کند در نتیجه خودرو به پارکینگی که ظرفیت کمتری دارد تغییر مسیر می‌دهد. این امر موجب تعادل بار و توزیع یکنواخت انرژی در پارکینگ‌ها می‌شود. خودروهای درخواست‌دهنده کار نیز اطلاعاتی از قبیل تعداد دستورالعمل هر درخواست و مهلت کار (حداکثر زمان پذیرفته شدن و انجام کار) را اعلام می‌کند. از آن جایی‌که خودروهای الکتریکی از نظر پردازنده متنوع‌اند با قرار گرفتن در پارکینگ علاوه‌بر میزان انرژی مازاد، توان پردازشی پردازنده خود را برحسب MIPS به کنترل‌کننده اعلام می‌کند. کنترل‌کننده با رسیدن درخواست‌های کار مدت زمان انجام کار(مدت زمانی‌که طول می‌کشد تا کار روی پردازنده موردنظر انجام شود) و میزان انرژی لازم برای انجام کار را تخمین می‌زند. هر زمان که درخواست کار به کنترل‌کننده برسد، الگوریتم تخصیص منابع اجرا می‌شود و کنترل‌کننده مناسب‌ترین منبع را برای پاسخ‌گویی به این درخواست انتخاب می‌کند. در شکل (۲) مراحل روش پیشنهادی به‌صورت خلاصه نشان داده شده است.

شکل (۲) : مراحل روش پیشنهادی

۱-۲- محاسبه انرژی مازاد خودروهای الکتریکی

به‌طور میانگین میزان مصرف انرژی خودروهای الکتریکی ۳۰ کیلووات ساعت در ۱۰۰ کیلومتر [20] می‌باشد. در این تحقیق به‌صورت کلی میزان انرژی لازم به ازای هر کیلومتر ۰.۳ کیلووات ساعت در نظر گرفته شده است.

در روش پایه، پارکینگ‌ها فاقد ایستگاه شارژ هستند و خودروهای منابع با رسیدن به پارکینگ انرژی مازاد خود را با درنظرگرفتن انرژی لازم برای رسیدن به مقصد، به کنترل‌کننده اعلام می‌کنند که در این حالت ممکن است با رسیدن خودروی الکتریکی به مقصد سطح انرژی آن به صفر برسد. در راه‌کار پیشنهادی نحوه محاسبه انرژی مازاد خودروهای منابع در دو حالت بررسی می‌شود:

حالت اول: با قرار گرفتن خودرو در پارکینگ، کنترل‌کننده موقعیت تمام ایستگاه‌های شارژ موجود در نقشه را بررسی می‌کند و با استفاده از الگوریتم مسیریابی دایجسترا کوتاه‌ترین مسیر بین مقصد خودرو و ایستگاه شارژ را پیدا می‌کند و میزان انرژی برای پیمودن این مسیر را تخمین می‌زند و در نتیجه علاوه‌بر میزان انرژی برای رسیدن به مقصد، میزان انرژی لازم برای رسیدن خودرو به نزدیکترین ایستگاه را از انرژی فعلی آن کسر می‌کند (روش پایه بهبود یافته). علت بهبود این روش نسبت به روش پایه این است که خودرو انرژی موردنیاز برای رسیدن به نزدیکترین ایستگاه شارژ را دارد و قبل از رسیدن به ایستگاه، شارژ آن تمام نمی‌شود.

حالت دوم: در حالت اول میزان انرژی که در اختیار شبکه گذاشته می‌شود نسبت به روش پایه کاهش پیدا می‌کند زیرا میزان انرژی لازم برای رفتن به ایستگاه شارژ هم از انرژی فعلی خودرو کسر می‌شود که این موضوع کاهش نرخ پاسخ موفق (نسبت تعداد درخواست‌های انجام شده توسط خودروهای منابع به کل درخواست‌ها) و توان عملیاتی شبکه را به دنبال دارد. بنابراین جهت بهبود این حالت طرح تشویقی در نظر گرفته می‌شود. به این صورت که پارکینگ‌ها مجهز به ایستگاه شارژ شوند و خودرو با رسیدن به پارکینگ میزان انرژی فعلی خود را بدون توجه به انرژی لازم برای رسیدن به مقصد و ایستگاه شارژ، به کنترل‌کننده اعلام کند. کنترل‌کننده اطلاعات لازم را جمع‌آوری می‌کند و تخصیص منابع را طبق راه‌کار پیشنهادی انجام می‌دهد. در نهایت با توجه به میزان انرژی که از خودرو کسرشده است به همان اندازه می‌توانند باتری خودرو را رایگان شارژ کنند. این حالت باعث افزایش میزان انرژی در شبکه می‌شود و از طرف دیگر با ارائه این طرح تشویقی تمایل افراد برای در اختیار گذاشتن منابع خودروهایشان به دلیل جبران آن افزایش می‌یابد.

استفاده از باتری خودروهای الکتریکی به‌طور مکرر باعث افزایش چرخه شارژ/ تخلیه می‌شود و این امر بر وضعیت سلامت باتری‌ها تأثیر می‌گذارد و باعث کاهش طول عمر آن‌ها شود. بنابراین باید خودرویی جهت انجام درخواست انتخاب شود که سطح سلامت (نسبت ظرفیت فعلی باتری خودرو به ظرفیت نامی اولیه) و سطح انرژی (میزان شارژ باتری نسبت به ظرفیت کل آن) بالاتری دارد[5]. در روش پیشنهادی سابقه خودروهای الکتریکی که منابع خود را در اختیار شبکه می‌گذارند در کنترل‌کننده ذخیره می‌شود و در اولین مراجعه خودرو به پارکینگ سطح سلامت و انرژی آن سنجیده و ذخیره می‌شود. سپس برای دفعات بعدی این دو مولفه مجدد ارزیابی و بروزرسانی می‌شود. با دریافت درخواست، کنترل‌کننده خودروهای منابعی که از سطح سلامت و انرژی بالاتری برخودار باشند انتخاب می‌کند.

۲-۲- راهکارپیشنهادی اولویت‌بندی درخواست‌ها

خودروهای منابع و خودروهای دارای کار با ایستادن در پارکینگ، درخواست خود را به همراه اطلاعات به کنترل‌کننده اعلام می‌کند. کنترل‌کننده با دریافت درخواست اعلام آمادگی خودرو برای در اختیار گذاشتن منابع، اطلاعات آن را در جدول منابع ذخیره می‌کند و سپس جدول کارها را چک می‌کند. اگر در جدول کار، چند درخواست وجود داشت منبع به کار با مهلت کمتر تخصیص داده می‌شود. اگر درخواستی موجود نبود وضعیت منبع آزاد می‌ماند تا درخواست کار جدیدی برسد. با رسیدن در خواست کار کنترل‌کننده جدول منابع موجود و آزاد در شبکه را بررسی می‌کنند و کمترین منبع را برای انجام این کار ضمن حفظ مهلت زمانی آن انتخاب می‌کند. این کار باعث ذخیره منابع بزرگ برای پذیرش درخواست‌های بزرگ در آینده می‌شود. اگر منابع موجود توانایی پاسخ‌گویی کار را داشته باشند کار پذیرفته می‌شود. در غیر این‌صورت کار منتظر می‌ماند تا منبع جدیدی برسد یا منبع اشغال آزاد شود (روش پیشنهادی).

در روش پیشنهادی فرض شده است که خودروهای منابع تنها یک درخواست را می‌توانند پذیرش کنند و در صورتی‌که درخواستی در حال انجام باشد و درخواست با مهلت زمانی کمتری برسد، نمی‌توانند کار در حال انجام را متوقف کنند تا به کار جدید با مهلت زمانی کمتر رسیدگی کنند. جهت بهبود روش پیشنهادی، خودروها دارای صف انتظار هستند. هنگامی که کار با مهلت زمانی کمتر می‌رسد و منابع مشغول هستند، کنترل‌کننده توسط تابع مکث و بازیابی منابع مشغول را بررسی می‌کند، اگر با پذیرش درخواست کار جدید، مهلت زمانی درخواست در حال انجام منقضی نشود و خودروی منبع انرژی لازم برای انجام درخواست را داشته باشد، کار درحال انجام به صف انتظار منتقل می‌شود و کار با اولویت بالاتر پذیرش و انجام می‌شود. بعد از اتمام درخواست جدید، درخواست منتظر از صف بازیابی می‌شود و ادامه روال انجام آن، طی می‌شود (راه‌کار پیشنهادی بهبود یافته).

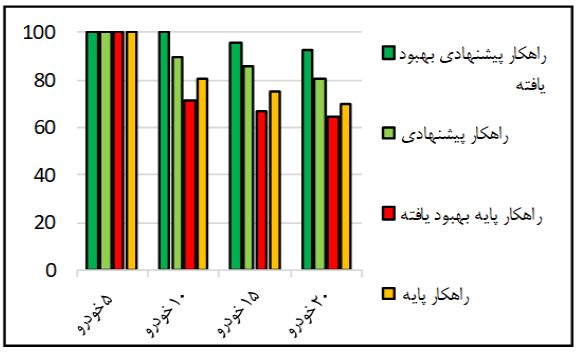
۳- شبیه‌سازی و تحلیل

با توجه به نوع ارتباطات و تحرک گره‌ها در این نوع از شبکه، از شبیه‌ساز OMNeT++ استفاده شده است. OMNeT++ یک کتابخانه و چارچوب شبیه‌سازی متن‌باز مبتنی بر مؤلفه، گسسته رخداد است که به زبان C++ پیاده‌سازی شده است. باتوجه به اینکه گره‌های موجود در شبکه خودروها هستند بنابراین برای شبیه‌سازی مدل‌های حرکتی و ترافیک شهری از نرم‌افزار SUMO استفاده شده است. SUMO یک شبیه‌ساز متن‌باز برای ترافیک است که با استفاده از آن می‌توان جاده‌ها، تقاطع‌ها، خودروها و جریان ترافیک را پیاده‌سازی نمود. برای پیاده‌سازی لایه کنترل و داده، از واسط SDVN استفاده می‌شود.

در این پژوهش فرض شده است خودروها دارای سیستم موقعیت یاب‌جهانی و تجهیزات OBU (برای ارسال/دریافت داده‌ها به خودروهای دیگر و یا واحدهای کنار جاده‌ای) هستند. واحد کنار جاده‌ای جهت انتقال داده‌ها به خودروها، پردازش و ارسال داده‌ها مستقر شده‌اند و همچنین جاده‌ها مجهز به زیرساخت‌های مورد نیاز جهت دریافت اطلاعات در تقاطع‌ها می‌باشند.

به منظور مقایسه راه‌کار پیشنهادی با دیگر سناریوهای مدنظر، پارامترهای نرخ پاسخ موفق، زمان انتظار کارها، گذردهی و بهره‌وری منابع در سناریوهای مختلف مورد ارزیابی قرار گرفته است. عملکرد هر سناریو با ظرفیت متفاوت پارکینگ، ترافیک‌های خودرویی متفاوت و تغییر در زمان‌بندی چراغ‌های راهنمایی موجود در نقشه مورد ارزیابی قرار گرفته است. جهت حصول نتایج قابل اتکاتر شبیه‌سازی شبکه مدنظر با پارامترهای مدنظر، ۵ بار تکرار شده‌است و در نهایت میانگین نتایج به دست آمده بیان شده‌است.

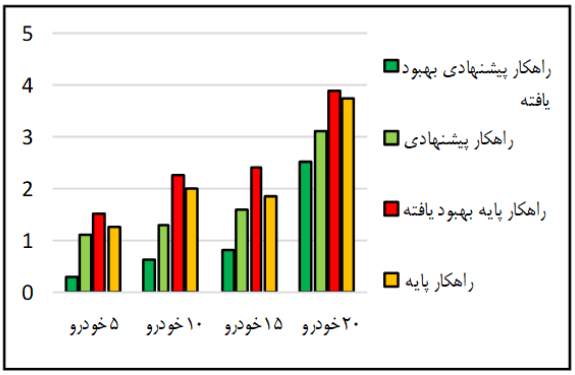
نرخ پاسخ موفق: خودروهای درخواست‌دهنده کار، با رسیدن به پارکینگ درخواست خودرو را به کنترل‌کننده اعلام می‌کنند. هنگامی‌که خودروی منبعی درخواست کار را قبول کرد، کنترل‌کننده یک پیام پذیرش به خودروی درخواست‌دهنده کار ارسال می‌کند. در این تحقیق نرخ پاسخ موفق در واقع نسبت کارهای انجام شده به کل کارهای ارسال شده توسط خودروهای درخواست‌دهنده کار است (شکل۳).



شکل (۳) : نرخ پاسخ موفق

در سناریوی راه‌کار پایه و پیشنهادی درخواست کار با مهلت کمتر باید در صف منتظر بماند تا منابع آزاد شوند و سپس منبع به آن تخصیص داده می‌شود بنابراین متحمل تاخیر می‌شود. اما در راه‌کار پیشنهادی بهبودیافته درخواست با مهلت کمتر بلافاصله پذیرفته می‌شود. علت کاهش نرخ پاسخ موفق در راه‌کار پایه بهبود یافته، کاهش میزان انرژی است که توسط خودروها در اختیار شبکه گذاشته می‌شود. در نتیجه نرخ پاسخ کاهش و تاخیر افزایش می‌یابد.

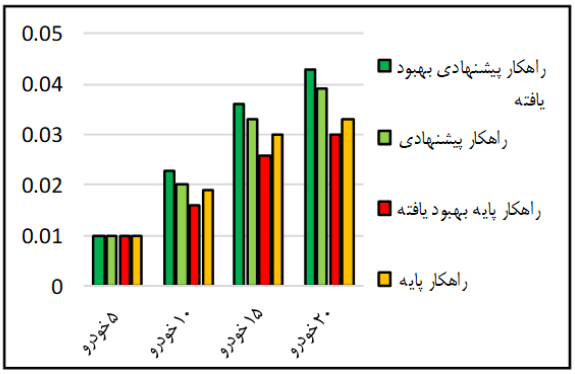
زمان انتظار کارها: زمان انتظار کار در واقع مدت زمانی که درخواست کار در صف منتظر می‌ماند تا منبعی به آن تخصیص داده شود و کار انجام شود. در شکل(۴) میانگین زمان انتظار در سناریوهای مورد ارزیابی نشان داده شده است.



شکل (۴) : میانگین زمان انتظار

با افزایش تعداد خودروها در پارکینگ تعداد درخواست‌های موجود در پارکینگ بیشتر می‌شود و با توجه به منابع موجود در پارکینگ ممکن است برخی از درخواست‌ها در صف منتظر بماند تا منبع به آن تخصیص داده‌شود. با این وجود در راه‌کار پیشنهادی با وجود طرحِ تشویقی منابعِ بیشتری در شبکه موجود است در نتیجه زمان انتظار درخواست‌ها کمتر از دوسناریوی قبلی می‌باشد. هم‌چنین در راه‌کار پیشنهادی بهبود یافته رسیدگی به درخواست‌ها با مهلت کمتر میزان زمان انتظار نسبت به راه‌کار پیشنهادی کمتر شده‌است.

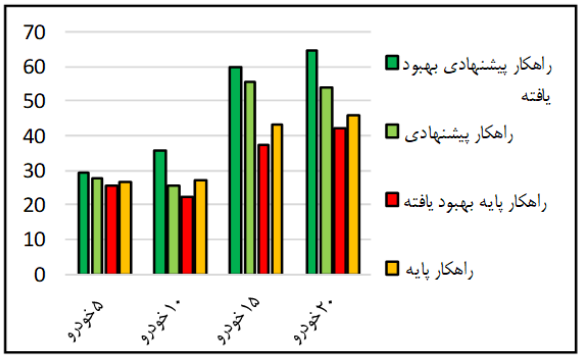
گذردهی: به تعداد کارهایی که در واحد زمان تکمیل می‌شوند گذردهی می‌گویند. در شکل (۵) گذردهی در سناریوهای مورد ارزیابی نشان داده شده است.



شکل (۵) : گذردهی

راه‌کار زمان‌بندی ارائه شده در این تحقیق به‌همراه طرح تشویقی با افزایش تعداد خودروهای موجود در پارکینگ عملکرد بهتری نسبت به روش پایه داشته است.

بهره‌وری منابع: خودروهای منابع مدت زمان معینی در پارکینگ قرار می‌گیرند. با توجه به این مدت زمان محدود باید راه‌کار زمان‌بندی به‌گونه‌ای باشد که از منابع آن‌ها به بهترین شکل ممکن استفاده کرد. در شکل (۶) بهره‌وری منابع در سناریوهای مورد ارزیابی نشان داده شده است.



شکل (۶) : بهره‌وری منابع

نتایج نشان می‌دهد راه‌کار پیشنهادی و بهبود آن با افزایش تعداد خودروها و درخواست‌ها عملکرد بهتری نسبت به روش پایه داشته است و درصد بهره‌وری از منابع بیشتر از سایر سناریوها می‌باشد. زیرا با اولویت‌بندی درخواست‌ها به‌همراه طرح تشویقی در نظر گرفته‌شده کارهای بیشتری توسط خودروهای منابع پذیرش و انجام می‌شوند و این موضوع افزایش نرخ پاسخ موفق و گذردهی را در شبکه به‌دنبال دارد.

۴- نتیجه‌گیری

با توجه به افزایش روز افزون تعداد خودروهای الکتریکی به‌همراه بالا رفتن قابلیت‌های پردازشی، ذخیره‌سازی و محاسباتی، این خودروها مدت زمان زیادی را در حالت پارک به‌سر می‌برند و منابع آن‌ها بلا استفاده می‌ماند. بنابراین می‌توان خودروهای الکتریکی پارک شده را به‌عنوان سرورهای محاسباتی(گره‌های مه) درنظرگرفت و از منابع کم مصرف آن‌ها برای خدمات محاسباتی استفاده کرد. این تحقیق به ارائه راه‌حلی برای اولویت‌بندی درخواست‌ها با مهلت زمانی محدود و یک طرح تشویقی برای افزایش میزان انرژی موجود در شبکه پرداخته شده‌است.

این مقاله در چهار بخش شامل مقدمه، روش پیشنهادی، شبیه‌سازی و نتیجه‌گیری تهیه شده‌است. نتایج شبیه‌سازی نشان می‌دهد که روش پیشنهادی با افزایش تعدادخودروهای موجود در پارکینگ عملکرد بهتری از خود نشان داده و باعث بهبود ۱۵.۶۹ درصدی نرخ پاسخ موفق و ۱۱.۰۲ درصدی بهره‌وری منابع و کاهش۱.۱۴ ثانیه‌ای میانگین زمان تاخیر درخواست‌ها در مقایسه با راه‌کار پایه است. ارائه یک مدل پیش‌بینی منابع، استفاده از الگوریتم‌های هوش مصنوعی جهت تخصیص بهتر و بهینه‌تر منابع، افزایش صف انتظار خودروها از یک درخواست به چند درخواست و مدیریت آن با رعایت مهلت زمانی درخواست‌ها از پیشنهادهای آتی ما است.

مراجع

[1] Al-Sarawi, S., Anbar, M., Abdullah, R., Al Hawari, A.B., *Internet of things market analysis forecasts*, 2020–2030. in 2020 Fourth World Conference on smart trends in systems, security and sustainability (WorldS4). 2020. IEEE.

[2] IDC, *IoT Growth Demands Rethink of Long-Term Storage Strategies, Says IDC.* https://www. idc. com/getdoc. jsp? containerId= prAP46737220, 2020.

[3] Gasmi, K., Dilek, S., Tosun, S., Ozdemir, *S., A survey on computation offloading and service placement in fog computing-based IoT.* The Journal of Supercomputing, 2022. 78(2): p. 1983-2014.

[4] Huang, C., Lu, R., K.-K.R. Choo, *Vehicular fog computing: architecture, use case, and security and forensic challenges.* IEEE Communications Magazine, 2017. 55(11): p. 105-111.

[5] Birhanie, H., *Resource Allocation in Vehicular Fog Computing for an Optimal Use of EVs Electric Vehicles Energy.* 2019, Université Bourgogne Franche-Comté.

[6] Zhang, Y., Wang, C.-Y., Wei, H.-Y., *Parking reservation auction for parked vehicle assistance in vehicular fog computing.* IEEE Transactions on Vehicular Technology, 2019 .(4)68 :p. 3126-3139.

[7] Rahman, F.H., Iqbal, A.Y.M., Newaz, S.S., Wan, A.T., Ahsan, M.S., *Street parked vehicles based vehicular fog computing: Tcp throughput evaluation and future research direction*. in 2019 21st International Conference on Advanced Communication Technology (ICACT). 2019. IEEE.

[8] Ning, Z., Huang, J., Wang, X., *Vehicular fog computing: Enabling real-time traffic management for smart cities.* IEEE Wireless Communications, 2019. 26(1): p. 87-93.

[9] Bhatia, J., Modi, Y., Tanwar, S., Bhavsar, M., *Software defined vehicular networks: A comprehensive review.* International Journal of Communication Systems, 2019. **32**(12): p. e4005.

[10] Zhou, Z., Liu, P., Zhang, Y., Mumtaz, S., Rodriguez, J., *Computation resource allocation and task assignment optimization in vehicular fog computing: A contract-matching approach.* IEEE Transactions on Vehicular Technology,(4)68.2019 :p. 3113-3125.

[11] Yang, M., Liu, N., Zuo, L., Gong, H., Liu, M., *Mobile parking incentives for vehicular networks: a deep reinforcement learning approach.* CCF Transactions on Pervasive Computing and Interaction, 2020. 2(4): p. 261-274.

[12] Bhardwaj, S., Iyer, S., Desai, T., Tumuluru, V.K., *Energy Scheduling and Computation Offloading for Building Operator using Parked Electric Vehicles*. in 2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS). 2020. IEEE.

[13] Le, T.H.T., Tran, N.H., Tun, Y.K., Kim, O.T.T., Kim, K., Hong, C.S., *Sharing incentive mechanism, Task assignment and resource allocation for task offloading in vehicular mobile edge computing*. in NOMS 2020-2020 IEEE/IFIP Network Operations and Management Symposium. 2020. IEEE.

[14] Nazih, O., Benamar, N., Addaim, A., *An incentive mechanism for computing resource allocation in vehicular fog computing environment.* in 2020 International Conference on Innovation and Intelligence for Informatics, Computing and Technologies (3ICT). 2020. IEEE.

[15] Alizadeh, M.R., Khajehvand, V., Rahmani, A.M., Akbari, E., *Task scheduling approaches in fog computing: A systematic review.* International Journal of Communication Systems, 2020. 33(16): p. e4583.

[16] Bittencourt, L.F., Diaz-Montes, J., Buyya, R., Rana, O.F., Parashar, M., *Mobility-aware application scheduling in fog computing.* IEEE Cloud Computing, 2017. 4(2): p. 26-35.

[17] Zhu, C., Tao, J., Pastor, G., Xiao, Y., Ji, Y., Zhou, Q., *Folo: Latency and quality optimized task allocation in vehicular fog computing.* IEEE Internet of Things Journal, 2018. **6**(3): p. 4150-4161.

[18] Auluck, N., Azim, A., Fizza, K., *Improving the schedulability of real-time tasks using fog computing.* IEEE Transactions on Services Computing, 2019.

[19] Zhou, Y., Liu, K., Xu, X., Guo, S., Wu, S., Lee, V., Son, S., *Distributed scheduling for time-critical tasks in a two-layer vehicular fog computing architecture.* in 2020 IEEE 17th Annual Consumer Communications & Networking Conference (CCNC). 2020. IEEE.

[20] Aksoy, L. *The impact of electric vehicles on electricity consumption.* in 2009 International Conference on Electrical and Electronics Engineering-ELECO 2009. 2009. IEEE.

پانویس ها

[1] Internet of Things
[2] Fog Computing
[3] On-Board Unit
[4] Software Defined Networking
[5] Million instructions per second

پنجمین کنفرانس ملی انفورماتیک ایران
پژوهشگاه دانشهای بنیادی، پردیس فرمانیه، تهران
۱۳ و ۱۴ دی ماه ۱۴۰۲

بیشینه سازی انتشار در شبکه های اجتماعی با استفاده از الگوریتم ژنتیک

محسن قنبری قمصری[۱]،سید مهدی وحیدی پور [۲]، فرشته دهقانی[۳]

[۱] دانشکده برق و کامپیوتر، دانشگاه کاشان، کاشان
mohsenghq@gmail.com

[۲] استادیار، دانشکده برق و کامپیوتر، دانشگاه کاشان، کاشان
vahidipour@kashanu.ac.ir

[۳] استادیار، دانشکده برق و کامپیوتر، دانشگاه کاشان، کاشان
fdehghani@kashanu.ac.ir

چکیده

با گسترش استفاده از شبکه‌های اجتماعی تحلیل آن‌ها هر روز دشوارتر می‌شود. یکی از مهمترین مسائل در تحلیل شبکه‌ها بیشینه سازی انتشار است. هدف اصلی بیشینه سازی انتشار، یافتن زیرمجموعه‌ای از افراد تأثیرگذار در شبکه می‌باشد، به نحوی که بتوانند تأثیرگذاری را در شبکه تحت یک مدل انتشار به حداکثر برسانند. بیشنیه سازی انتشار به دلیل کاربردهای مختلف نظیر توصیه محصولات، بازاریابی، انتشار اطلاعات و ایمن سازی بیماری توجه زیادی را به خود جلب کرده است. معمولاً برای حل این مسئله آن را به صورت یک مسئله بهینه‌سازی گسسته مدل می‌کنند و از الگوریتم‌های تخمینی یا فرا ابتکاری برای حل آن استفاده می‌کنند. با این حال به سختی می‌توان بین بهینگی زمانی و دقت تعادل برقرار کرد. در این مقاله این مسئله توسط یک الگوریتم فراابتکاری حل شده و توانسته است علاوه بر حفظ دقت تا حدود ۴۰ درصد زمان اجرا را کاهش دهد.

کلمات کلیدی

شبکه‌های پیچیده، بیشینه سازی انتشار، الگوریتم ژنتیک، بهینه سازی، شبکه‌های اجتماعی

۱- مقدمه

در دهه اخیر، تحقیقات زیادی در شبکه‌های اجتماعی صورت گرفته است. این شبکه‌ها ساختار نامنظم و پیچیده دارند. شبکه اجتماعی شامل ساختاری است که افراد یا سازمان‌ها و روابط بین آنها را در بر می‌گیرد. بر اساس نوع شبکه، روابط بین اجزای تشکیل دهنده می‌تواند نشان دهنده دوستی، همکاری، روابط علمی، دنبال کردن و غیره باشد. شبکه‌های اجتماعی افراد زیادی را در مدت زمان کوتاهی به یکدیگر متصل می‌کنند. این شبکه‌ها روابط جدیدی را بین مردم ایجاد کرده اند. از طریق تعامل در شبکه‌های اجتماعی، داده‌های زیادی رد و بدل می‌شود، ایده‌ها و دانش به اشتراک گذاشته می‌شود و افراد بر یکدیگر تأثیر می‌گذارند. در واقع، حجم وسیعی از داده‌ها به دلیل توسعه و انتشار شبکه‌های اجتماعی تهیه شده است. تجزیه و تحلیل این داده‌ها می‌تواند به افراد در درک ویژگی‌های ساختاری و رفتاری شبکه‌های اجتماعی کمک کند و به بسیاری از سوالات اقتصادی، اجتماعی و جامعه شناختی پاسخ دهد. بنابراین تحلیل شبکه‌های اجتماعی به عنوان یکی از زیر شاخه‌های علوم کامپیوتر در سال‌های اخیر بسیار مورد توجه قرار گرفته است. از جمله موضوعات چالشی در تحلیل شبکه‌های اجتماعی می‌توان به نمونه‌گیری شبکه[۱] [۱]، پارتیشن بندی شبکه[۲] [۲] و تشخیص ناهنجاری[۳] [۳] اشاره کرد.

یکی از مهمترین مسائل در تحلیل شبکه‌ها بیشینه سازی انتشار[۴] است. هدف اصلی بیشینه سازی انتشار، یافتن زیرمجموعه‌ای از افراد تأثیرگذار است که می‌توانند تأثیرگذاری را در شبکه تحت یک مدل انتشار به حداکثر برسانند. در تئوری گراف، تأثیر معیاری است که نشان می‌دهد یک گره چقدر می‌تواند وضعیت دیگر گره‌های یک گراف را تغییر دهد. مطالعه بیشینه‌سازی انتشار متکی بر مدل‌های انتشار و الگوریتم‌های شناسایی افراد تأثیرگذار است.

در این مقاله از روش‌های تکاملی برای حل مسئله بیشینه‌سازی انتشار استفاده شده است. نسل‌های مختلف از جواب‌ها، با استفاده از شبیه‌سازی انتشار ارزیابی شده و درنهایت بهترین جواب گزارش می‌شود. استفاده از الگوریتم ژنیتک در حل این مسئله در کارهای پژوهشی دیده می‌شود. در این مقاله، استفاده از ذخیره‌سازی اطلاعات با استفاده از توابع درهم‌ساز پیشنهاد می‌شود که علاوه بر حفظ دقت، باعث افزایش ۴۰٪ در سرعت الگوریتم خواهد شد. همچنین روش تکاملی جدید LPB نیز با استفاده همزمان از ذخیره‌سازی اطلاعات در حل مسئله بیشینه‌سازی انتشار استفاده می‌شود که نتایج آن با روش ژنتیک مقایسه شده است.

[1] Network sampling
[2] Network partitioning
[3] Anomaly detection
[4] Influence Maximization

ادامه این مقاله به شرح زیر سازماندهی شده است: بخش دوم شامل مفاهیم استفاده شده در این مقاله است. در بخش سوم، شرحی از کارهای مرتبط در زمینه بیشینه سازی انتشار در شبکه‌های اجتماعی بررسی می‌شود. روش پیشنهادی در بخش چهارم شرح داده شده است. در بخش پنجم، آزمایش‌های انجام شده قرار دارد. در نهایت در قسمت ششم نتایج گزارش شده است.

۲- مفاهیم پایه

در این بخش بیشینه‌سازی انتشار و یک مدل انتشار رایج مرور می‌شود. سپس دو الگوریتم ژنتیک و LPB که در این مقاله استفاده شده‌اند بررسی می‌شوند.

۲-۱- بیشینه سازی انتشار

بیشینه سازی انتشار یک مسئله بهینه‌سازی است که هدف آن پیدا کردن مجموعه گره‌های موثر در یک گراف است به طوری که اگر انتشار از آن‌ها شروع شود، بتوانند درنهایت بیشترین تعداد گره‌ها را فعال کنند. در حالت کلی گره‌ها دو وضعیت دارند: غیرفعال و فعال. اگر غیرفعال باشند یعنی هنوز تأثیر نگرفته و اگر فعال باشند یعنی تحت تأثیر یکی از همسایه‌های خود تغییر وضعیت داده‌اند. به عبارت دیگر هدف یافتن زیرمجموعه کوچکی از k گره از یک شبکه به منظور حداکثر کردن تعداد کل گره‌های تحت تأثیر این k گره است.

اگر گراف $G(V,E)$ را داشته باشیم که V مجموعه گره‌های گراف و E مجموعه یال‌ها باشد، هدف پیدا کردن مجموعه S است که $S \subseteq V$ و $|S| = k$ باشد و k تعداد گره‌هایی است که به دنبال آن هستیم تا $S = \underset{S}{\operatorname{argmax}} f(S)$ برقرار شود و f یک مدل انتشار است.

مجموعه S را مجموعه گره‌های بذر[5] هم می‌گویند. در روند یافتن گره‌های مدنظر ابتدا مجموعه بذر فعال می‌شوند و طبق مدل انتشار تلاش می‌کنند تا گره‌های همسایه خود را فعال کنند، هر گره‌ای که فعال شد نیز می‌تواند همسایه های خود را فعال کند و این کار تا زمانی که گره جدیدی فعال نشود ادامه پیدا می‌کند.

۲-۲- مدل Independent cascade

در تئوری گراف، از مدل‌های انتشار برای مدل‌سازی نحوه انتشار اطلاعات یا تأثیرگذاری گره ها از طریق شبکه استفاده می‌شود. مدل Independent cascade یا IC در شکل ۱ توضیح داده شده است. ورودی این تابع گراف G، مجموعه بذر A_0، تعداد شبیه سازی‌های مونت-کارلو[6] MC و احتمال-های تأثیر گره i روی گره j است که با λ_{ij} نشان داده شده است. خروجی تابع میانگین تعداد گره‌های فعال شده است. در این مدل، کل شبکه اجتماعی به عنوان یک گراف نشان داده می‌شود که در آن گره‌ها افراد و یال‌ها روابط آنها را نشان می‌دهند. برای شروع یک مجموعه A_0 شامل k گره فعال در نظر گرفته می‌شود. مجموعه A نشان دهنده گره‌هایی است که تا به حال فعال شده‌اند و در ابتدا همان A_0 است. در هر مرحله یک مجموعه B وجود دارد که شامل گره‌هایی است که خود فعال هستند و اجازه دارند همسایه‌های خود را فعال کنند، این فعال سازی با احتمال از پیش تعیین شده λ_{ij} که یا از دانش قبلی آمده است یا برای همه یال‌ها یکسان در نظر گرفته می‌شود، فقط یک بار می‌تواند اتفاق بیفتد. گره‌های فعال شده جدید در هر مرحله در مجموعه $nextB$ نگه داری شده و چون این گره‌های جدید مجاز هستند که همسایه های خود را فعال کنند داخل مجموعه B قرار می‌گیرند. سپس تمام گره‌های فعال شده جدید به مجموعه A اضافه می‌شود. تا زمانی که مجموعه B خالی نباشد یعنی گره فعال شده جدیدی داشته باشیم این مراحل تکرار می‌شود. چون این فرایند تصادفی است به تعداد MC که کاربر آن را تعیین می‌کند این کار تکرار می‌شود و در نهایت میانگین تعداد گره‌های فعال شده برگردانده می‌شود.

```
1   spread = 0
2   repeat:
3      τ ← False    τ: Has The Propagation Ended?
4      A ← A0       A: Active Nodes
5      B ← A        B: the set of nodes activated
6      while not τ do
7         nextB ← ∅;
8         for each n ∈ B do
9            for each neighbour m of n , where m ∉ A, do
10              with probability λij, add m to nextB
11        B ←nextB
12        A ←A ∪ B
13        if B is empty then
14           τ← True
15     Spread. Append(Len(A))
16  until MC
17  return mean(Spread)
```

شکل ۱: الگوریتم Independent cascade model

۲-۳- الگوریتم ژنتیک

الگوریتم ژنتیک از تکنیک‌های زیست‌شناسی مانند وراثت، جهش زیست‌شناسی، اصول انتخابی داروین و انتقال ویژگی در نسل‌های بعد برای یافتن فرمول بهینه جهت پیش‌بینی یا تطبیق الگو استفاده می‌شود. الگوریتم ژنتیک در شکل ۲ نشان داده شده است. به طور کلی ابتدا چند پاسخ تصادفی با نام کروموزوم[7] تولید می‌شود سپس با استفاده از عملگرهای الگوریتم ژنتیک آن ها را تغییر می‌دهیم تا به پاسخ های بهتری تبدیل شوند. ورودی الگوریتم یک مسئله بهینه‌سازی و خروجی آن بهترین پاسخ پیدا شده است. ابتدا جمعیت اولیه‌ای از کروموزومها به صورت تصادفی ساخته می‌شود (فراخوانی تابع CreatInitialPopulation()). جمعیت اولیه با فراخوانی تابع Evaluate() ارزیابی می‌شود؛ میزان خوب بودن آن تعیین می‌شود. سپس به تعداد نسل Number of Generation که به دلخواه تعیین می‌شود، جمعیت جدید از روی جمعیت قبلی ساخته می‌شود.

برای ساخت جمعیت جدید مراحل ۵ تا ۱۰ مشخص شده شکل ۲ انجام می‌شود. در مراحله ۵، کروموزومهایی به عنوان والدین انتخاب می‌شوند. از ترکیب دو والد در مرحله ۶ (یعنی Crossover()) دو فرزند جدید که عضو جمعیت بعدی هستند ساخته می‌شوند. تحت عملگر جهش، تعدادی از فرزندان جدید دچار تغییرات اندکی می‌شوند. فرزندان جدید در جمعیت جدید قرار می‌گیرند و مجدداً ارزیابی خواهند شد. اما یک اصل در الگوریتم ژنتیک،

[5] Seed
[6] Monte-Carlo
[7] Chromosome

نخبه‌گرایی[8] است که در آن کروموزومهایی که در نسل قبلی جوابهای خوبی را نمایش می‌دهند، مستقیم و بدون هیچ تغییری در جمعیت جدید قرار می‌گیرند (مرحله ۱۱ در شکل ۲)

```
1   InitializationPopulation ()
2       CreateInitialPopulation ()
3       Evaluate ()
4   repeat
5       Select Parents ()
6       Crossover ()
7       Mutation ()
8       Replacement ()
9       Evaluate ()
10      Elitism ()
11  until Number of Generations
12  return Best Solution
```

شکل ۲: الگوریتم ژنتیک

۴-۲- الگوریتم LPB

در [۴] روش بهینه سازی جدیدی با نام LPB[9] معرفی شده است که شبیه الگوریتم ژنتیک انتخاب جمعیت اولیه تصادفی، تقاطع و جهش دارد. ایده اصلی آن مبتنی بر فارغ التحصیلی دانش آموزان دبیرستانی است که می‌خواهند به دانشگاه بروند. به عنوان اولین گام در الگوریتم، به طور تصادفی جمعیتی از دانش آموزان فارغ التحصیل M ایجاد می‌شود که می‌خواهند برای بخش‌های مختلف در دانشگاه‌های مختلف درخواست دهند. هر بخش دانشجوهایی را می‌پذیرد که معدل بالاتر یا مساوی با حداقل معدل مورد نیاز دارند. برای نشان دادن این موضوع در الگوریتم، ابتدا از پارامتر dp برای انتخاب تصادفی از عناصر M استفاده می‌شود. سپس برازندگی هر فرد محاسبه شده و مرتب می‌شود. سپس با توجه به برازندگی آنها را به دو گروه خوب و بد تقسیم می‌کنیم. اولی شامل افرادی است که معدل بالاتری دارند و گروه بد شامل بقیه است. پس از این، برازندگی افراد در جمعیت اصلی M محاسبه می‌شود. آن دسته از افرادی که دارای برازندگی کمتر یا برابر با بالاترین برازندگی در جمعیت بد هستند، به جمعیت بد منتقل می‌شوند. بقیه افراد به دو گروه تقسیم می‌شوند. کسانی که برازندگی کمتر یا برابر با بالاترین برازندگی (بهترین برازندگی) در جمعیت خوب دارند به جمعیت خوب منتقل می‌شوند و کسانی که برازندگی بیشتر از بالاترین برازندگی در جمعیت خوب دارند به جمعیت کامل منتقل می‌شوند. سپس تعداد فراگیران مشخص شده توسط دانشگاه از بین جمعیت کامل و جمعیت خوب انتخاب می‌شود. اگر تعداد افراد در این دو جمعیت کمتر از تعداد دانشجویان مشخص شده توسط گروه بود، بهترین افراد از جمعیت بد اضافه می‌شوند.

دانشجویان می‌توانند بر رفتار یکدیگر تأثیر بگذارند. مثلا هنگامی‌که آنها به صورت گروهی کار می‌کنند یا زمانی که از یکدیگر کمک می‌خواهند. برای نشان دادن این امر در الگوریتم عملگر تقاطع از الگوریتم ژنتیک استفاده شده است. مثلا اگر یک دانشجو از دیگری کمک بخواهد مقداری از اطلاعات آن تغییر می‌کند و می‌توان آن را به عنوان تغییر در ژن‌ها تعبیر کرد.

[8] Elitism
[9] Learner performance-based behavior

اطلاعاتی که خود فرد خارج از جمعیت کسب می‌کند نیز در رفتار فرد تأثیر می‌گذارد. این مورد در الگوریتم با جهش اعمال می‌شود.

۳- کارهای مرتبط

مسئله بیشینه سازی انتشار اولین بار در [۵][۶] به عنوان یک مسئله برای یافتن مجموعه‌ای از مشتریان مفید در بازاریابی مطرح شد. برای مدلسازی این موضوع، آنها از فیلدهای تصادفی مارکوف استفاده کردند. اگرچه روش ارائه شده می‌تواند مشکل بیشینه‌سازی انتشار را حل کند، اما نمی‌تواند تأثیر کاربران بر یکدیگر را به وضوح شناسایی کند. مقالات [۷][۸] مسئله بیشینه‌سازی انتشار را به عنوان یک مسئله بهینه‌سازی گسسته برای اولین بار مدل کردند. آنها ثابت کردند که به طور کلی، بیشینه سازی انتشار جزو مسائل NP-Hard طبقه بندی می‌شود، اما آنها دو مدل انتشار به نام‌های Independent cascade mode و Linear threshold model و همچنین یک الگوریتم حریصانه با حد تقریب $1-1/(e-\varepsilon)$ پیشنهاد دادند که e و ε به ترتیب عدد اویلر و دقت تخمین هستند. این الگوریتم حریصانه هرچند دقت خوبی دارد اما مقیاس پذیر نیست. بعد از آن چند راهکار برای بهبود این الگوریتم ارائه شد که به عنوان مثال می‌توان از CELF[۹]،CELF++[۱۰]، TIM[۱۱]، BCT[۱۲]، RIS [۱۳] و MixedGreedy [۱۴] نام برد.

در برخی از کارهای قبلی، از روش‌های اکتشافی برای حل مسئله بیشینه‌سازی انتشار استفاده شد. برخی از این مدل‌ها بر اساس معیارهای مرکزیت مانند درجه، گره‌های موثر را تعیین می‌کردند [۱۲] [۱۱] ؛ آنها مستقل از مدل انتشار بودند. گروه دیگری از روش‌های اکتشافی به طور خاص برای یک مدل انتشار طراحی شده‌اند و کارایی مناسبی را در مدل پیشنهادی نشان دادند به عنوان مثال IRIE[۱۳] برای مدل انتشار Independent cascade mode به خوبی کار می‌کند. خوبی روش‌های اکتشافی این بود که هم پیچیدگی زمانی را بهبود و هم مقیاس پذیری را افزایش دادند.

در [۱۸] از الگوریتم ژنتیک ساده برای حل مسئله بیشینه‌سازی انتشار استفاده شده و نشان داده است که با استفاده از یک عملگر ساده ژنتیکی، می توان یک راه حل تقریبی برای تأثیرگذاری در زمان اجرای بهتر نسبت به استفاده از الگوریتم های حریصانه قبلی به دست آورد. در [۱۹]، با گسترش الگوریتم ژنتیک قبلی، یک الگوریتم فرا ابتکاری برای انتخاب یک زیر مجموعه با طول ثابت پیشنهاد کردند. برای این پیشنهاد از اطلاعات مربوط به ساختار شبکه استفاده شده است. در [۲۰] از الگوریتم ژنتیک برای یافتن راه‌حل و یک الگوریتم حریصانه برای بهبود راه‌حل به‌دست‌آمده از طریق الگوریتم ژنتیک به منظور افزایش کارایی مسئله بیشینه‌سازی انتشار استفاده می‌شود.

۴- روش پیشنهادی

فرض کنید گراف $G=(V,E)$ داده شده است که V مجموعه گره‌ها و E مجموعه یال‌هاست. همچنین k تعداد گره‌های بذر است. برای حل مسئله بیشینه‌سازی انتشار از مدل IC استفاده خواهد شد. از دید بهینه‌سازی هدف حل معادله (۱) است، که در آن $f(s)$ میانگین تعداد گره‌های فعال شده است که در رابطه (۲) معرفی شده است.

$$s = \underset{s}{\operatorname{argmax}} f(s) \qquad (۱)$$

$$f(s) = \frac{1}{M}\sum_{i=1}^{M} f_i(s) \quad (۲)$$

در رابطه (۲)، M تعداد شبیه سازی‌های مونت–کارلو است.

برای حل این مسئله بهینه‌سازی از الگوریتم ژنتیک استفاده شده است. به این صورت که ابتدا چند کروموزوم مانند شکل ۳ تشکیل می‌شود که هر ژن عدد یک گره مشخص را تعیین می‌کند و تعداد ژن‌ها همان k است. در شکل ۳، مجموعه گره‌های ۱، ۳، ۷، ۸ و ۱۱ به عنوان ۵ گره بذر پیشنهاد شده است $(k = 5)$.

۷	۳	۱۱	۸	۱

شکل ۳: نمونه یک کروموزوم ساخته شده

سپس تمام کروموزوم‌ها باید ارزیابی شوند که این کار توسط مدل انتشار IC صورت می‌گیرد. به این صورت که گره‌های موجود در کروموزوم در گراف فعال شده و ارزش آن کروموزوم که همان تعداد گره‌های فعال شده است با یک عدد بازمی‌گردد. هر کروموزومی که بیشترین گره را فعال کرده باشد شانس بالاتری برای انتخاب به عنوان والد خواهد داشت. همچنین از نخبه–گرایی نیز استفاده شده تا دقت الگوریتم بالاتر برود.

تا اینجا فقط از الگوریتم ژنتیک استفاده شد و مشکل اجرای طولانی برای گراف‌های بزرگ همچنان باقی است. در روش پیشنهادی از عملگرهای تقاطع و جهش الگوریتم ژنتیک استاندارد استفاده شده است و پیشنهاد این مقاله برای بهبود در زمان اجرا استفاده از تابع درهم ساز [۱۰] و حذف گره‌های کم اهمیت است. به این صورت که در هر بار اجرای الگوریتم IC برای یک کروموزوم مقدار نهایی محاسبه شده به همراه آرایه‌ای شامل همان کروموزوم در یک جدول ذخیر می‌شود و در مراحل بعدی اگر کروموزوم تکراری وارد شد به جای اجرای کامل تابع IC فقط مقدار آن از جدول بازگردانده می‌شود. نام این روش $GA-H$ است. برای این کار از جدول با اندازه $3*10^7$ سطر شامل جفت‌های کلید–مقدار برای ذخیره‌سازی میانگین ارزیابی هر کروموزوم استفاده می‌شود. هر کلید توسط تابع درهم‌ساز به یک کد تبدیل می‌شود و در جایگاه همان کد در جدول قرار می‌گیرد در این حالت موقع جستجو نیاز نیست کل جدول جستجو شود و تنها کافی است کلید را تبدیل به آدرس کرده و همان یک خانه بررسی شود و اگر مقداری در آن بود برگردانده شود.

برای بهبود زمانی بیشتر، الگوریتم $GA-D$ معرفی می‌شود که همان $GA-H$ است با این تفاوت که گره‌های با درجه کمتر از m از گراف حذف می‌شوند تا محاسبات در هر مرحله کاهش یابد. واضح است که گره‌هایی با درجه کمتر، همسایه‌های کمی دارند و حتی اگر تعدادی از آن‌ها را فعال کنند در جواب نهایی تأثیر زیادی ندارد چون همسایه‌های آن‌ها هنوز می‌توانند توسط گره‌های دیگری فعال شوند. برای یافتن مقدار مناسب برای حداقل درجه، آزمایشی در بخش آزمایش‌ها طراحی شده است که بر اساس آن پیشنهاد می‌شود که گره‌هایی با درجه کمتر از ۵ حذف شوند.

۵- آزمایش‌ها

در این بخش چهار آزمایش برای بررسی کارایی الگوریتم با روش‌های قبلی انجام شده است. مقایسه در زمان اجرا و دقت مد نظر است و منظور از دقت همان تعداد گره‌های فعال شده نهایی است.

جدول 1 نشان دهنده پارامترهای الگوریتم ژنتیک استفاده شده در آزمایش–هاست. در جدول 2 مشخصات مجموعه داده‌های مورد استفاده ذکر شده است. در ادامه این دو دیتاست مرور می‌شوند:

- **دیتاست Stackoverflow**: کاربران سوالات خود را در وب سایت Stack Overflow ارسال می‌کنند و از سایر کاربران پاسخ دریافت می‌کنند. کاربران ممکن است هم در مرحله پرسش و هم در مرحله پاسخگویی نظر بدهند و در نهایت یک شبکه به دست می آید.
- **دیتاست Email**: در یک سرور ایمیل، تعداد افرادی که از طریق ایمیل با هم در ارتباط هستند تشکیل یک شبکه می‌دهند.

جدول 1: پارامترهای الگوریتم ژنتیک

پارامتر	مقدار
تعداد جمعیت اولیه	۱۰۰
تعداد مراحل الگوریتم	K*۱۰۰
احتمال جهش	۰.۱
نوع تقاطع کروموزوم‌ها	تک نقطه‌ای
نسبت Elitism	۰.۰۲

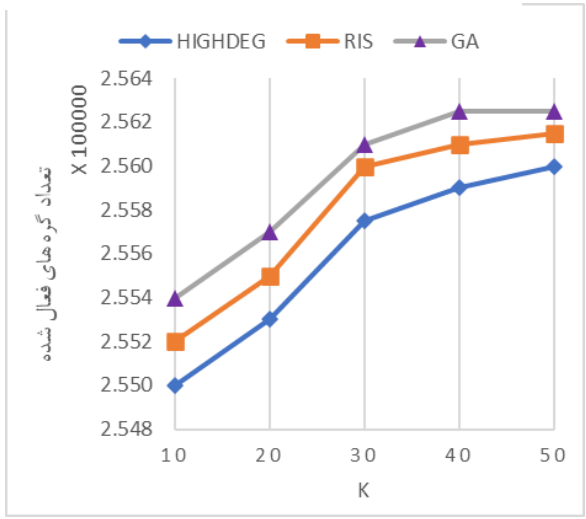
جدول 2: اطلاعات دیتاست‌ها

دیتاست	Stack overflow	Email
تعداد گره‌ها	۱۶۴۶۳۳۸	۱۱۳۳
تعداد یال‌ها	۱۱۳۷۰۳۴۲	۱۰۹۰۲

۵-۱- آزمایش اول

این آزمایش دقت الگوریتم پیشنهادی را نسبت به الگوریتم‌های HIGHDEG [۱۲] و RIS [۹] روی دیتاست stackoverflow مقایسه می‌کند. در شکل ۴ دقت سه الگوریتم با هم مقایسه شده است که GA الگوریتم ژنتیک است.

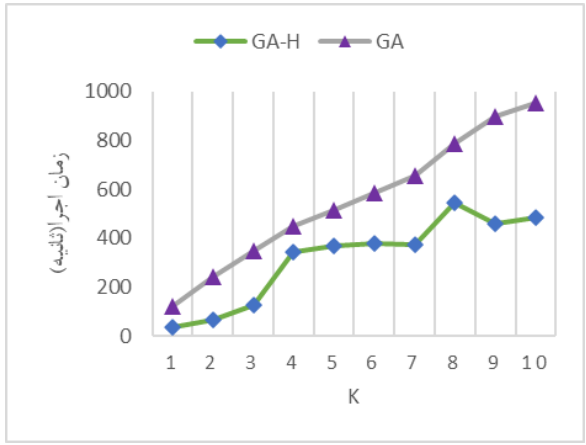
[10] Hash function



شکل 4: مقایسه دقت با الگوریتم‌های قبلی

۵-۲- آزمایش دوم: تأثیر استفاده از تابع درهم ساز

این بخش، با هدف تأثیر استفاده از تابع درهم سازی در زمان اجرای الگوریتم پیشنهادی طراحی شده است. همانطور که در شکل ۵ مشخص است اضافه کردن جدول و ذخیره مقادیر باعث می‌شود زمان اجرا تا ۴۰٪ بهبود یابد.

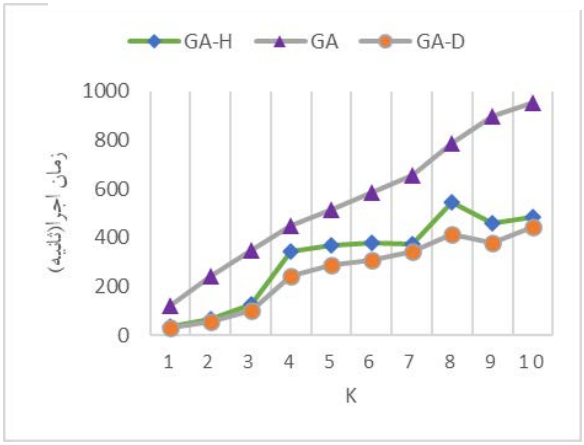


شکل ۵: تأثیر Hash function روی زمان اجرا

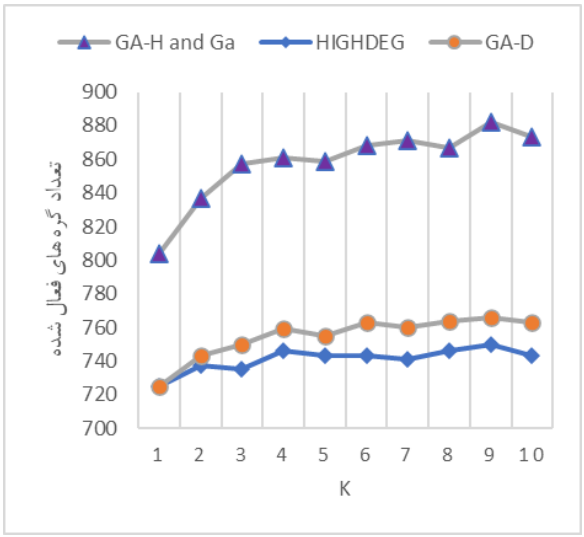
۵-۳- آزمایش دوم: تأثیر هرس کردن

در این آزمایش هدف مقایسه زمان اجرا و همچنین دقت سه نسخه از الگوریتم ژنتیک که در این مقاله معرفی شد به ازای گره‌های فعال اولیه است. در شکل ۷ زمان اجرا و در شکل ۸ دقت سه روش نشان داده شده مقایسه شده است.

در شکل ۶ مشخص است زمان اجرا حدود ۱۰ درصد نسبت به GA-H بهبود یافته است از طرف دیگر شکل ۷ نشان می‌دهد دقت به میزان ۱۰ درصد کاهش یافته است. اگر معیار حذف گره‌ها به جای درجه ۵ کمتر از ۵ باشد، میزان بهبود و کاهش دقت نیز کمتر می‌شود.



شکل ۶: تأثیر حذف گره‌ها روی زمان اجرا

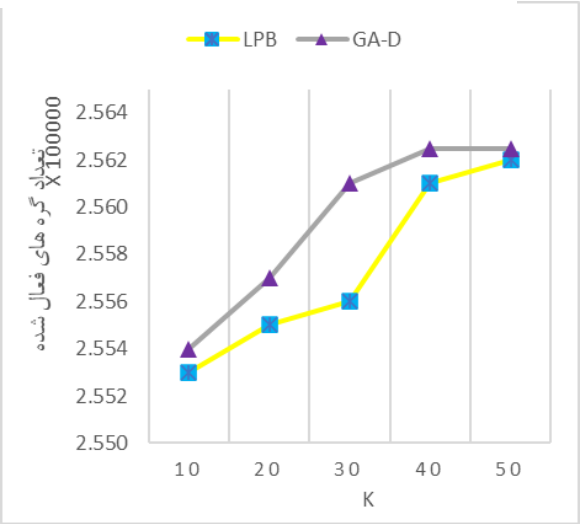


شکل ۷: تأثیر حذف گره‌ها روی دقت الگوریتم

۵-۴- بهبود روش پیشنهادی با استفاده از LPB

با توجه به آزمایش اول، مشخص است که استفاده از الگوریتم ژنتیک به زمان بسیار زیادی نیاز دارد. بنابراین در این آزمایش به دنبال پاسخ به این سوال هستیم که آیا می‌توان با اندکی کاهش در کیفیت پاسخ، نیاز به زمان کمتری داشته باشیم؟

در راستای پاسخ به این سوال، از الگوریتم LPB به جای الگوریتم ژنتیک استفاده می‌کنیم. شکل ۸ تعداد نودهای فعال شده را برای دو روش GA و LPB نشان می‌دهد. همچنین در جدول ۳ زمان این دو الگوریتم مقایسه شده است. نتایج نشان می‌دهد که با اجرای الگوریتم LPB در ازای مقداری کاهش دقت، مقدار زیادی در زمان اجرا بهبود حاصل شده است.



شکل ۸: مقایسه دقت الگوریتم LPB و GA

جدول 3: مقایسه زمان اجرای الگوریتم‌ها(ثانیه)

k	۵۰	۳۰	۱۰
GA-D	۲۷۱۴	۱۱۵۰	۵۰۹
LPB	۷۹۴	۳۵۰	۱۴۰

۶- نتیجه گیری

با توجه به اینکه اکثر شبکه‌های دنیای واقعی سایز بزرگ دارند تحلیل آن‌ها زمانبر است. در این مقاله سعی شد با پیشنهاد چند راهکار برای مسئله بیشینه سازی انتشار علاوه بر حفظ دقت در الگوریتم ژنتیک، زمان اجرا نیز بهبود یابد. در مرحله اول از تابع درهم‌ساز استفاده شد و بدون از دست دادن دقت، زمان اجرا به طور میانگین حدود ۴۰ درصد بهبود پیدا کرد. در مرحله بعد گره‌هایی که انتظار می‌رفت کمتر موثر هستند حذف شدند که حدود ۱۰ درصد در زمان اجرا موثر بود اما از طرفی دقت نیز به همین اندازه کم شد. درنهایت از یک الگوریتم تکاملی بهبود یافته LPB به جای الگوریتم ژنتیک استفاده شد و در ازای مقداری کاهش دقت، مقدار زیادی در زمان اجرا بهبود حاصل شد.

مراجع

[1] M. Granovetter, “Network sampling: Some first steps”, *American journal of sociology*, vol. 81, no. 6, pp. 1287-1303, 1976.

[2] L. A. Sanchis, “Multiple-way network partitioning”, *IEEE Transactions on Computers*, vol. 38, no. 1, pp. 62-81, 1998.

[3] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection: A survey”, *ACM computing surveys (CSUR)*, vol. 41, no. 3, pp. 1-58, 2009.

[4] C. Rahman and T. Rashid, “A new evolutionary algorithm: Learner performance-based behavior algorithm”, *Egyptian Informatics Journal,* vol. 22, no. 2, pp. 213-223, 2021.

[5] M. Richardson and P. Domingos, “Mining knowledge-sharing sites for viral marketing”, *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, pp. 61–70, 2002.

[6] P. Domingos and M. Richardson, “Mining the network value of customers”, Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, pp. 57–66, 2001.

[7] D. Kempe, J. Kleinberg, and É. Tardos, “Maximizing the spread of influence through a social network”, *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, pp. 137–146, 2003.

[8] D. Kempe, J. Kleinberg, and É. Tardos, “Influential nodes in a diffusion model for social networks”, *Lecture Notes in Computer Science*, 3580, pp. 1127–1138, 2005.

[9] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance, “Cost-effective outbreak detection in networks”, *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, pp. 420–429, 2007.

[10] Goyal, W. Lu, and L.V.S. Lakshmanan, “CELF++: optimizing the greedy algorithm for influence maximization in social networks”, *Proceedings of the 20th International Conference Companion on World Wide Web*, Hyderabad, India, 2011.

[11] Y. Tang, X. Xiao, and Y. Shi, “Influence maximization: Near-optimal time complexity meets practical efficiency”, *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*, pp. 75–86, 2014.

[12] H.T. Nguyen, M.T. Thai, and T.N. Dinh, “A billion-scale approximation algorithm for maximizing benefit in viral marketing”, IEEE/ACM Transaction Networks. Vol. 25, no. 4, pp. 2419–2429, 2017.

[13] C. Borgs, M. Brautbar, J. Chayes, and B. Lucier, “Maximizing social influence in nearly optimal time”, *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, SIAM, pp. 946–957, 2014.

[14] J. Lv, J. Guo, and H. Ren, “Efficient greedy algorithms for influence maximization in social networks”, *Journal of Information Processing Systems*, vol. 10, no. 3, pp. 471–482, 2012.

[15] L.C. Freeman, “Centrality in social networks conceptual clarification”, *Social Networks*, vol. 1, no. 3, pp. 215–239, 1978.

[16] W. Chen, Y. Wang, and S. Yang, “Efficient influence maximization in social networks”, Presented at the Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, 2009.

[17] K. Jung, W. Heo, and W. Chen, “Irie: Scalable and robust influence maximization in social networks”, *the 12th International Conference on Data Mining*, IEEE, pp. 918–923, 2012.

[18] D. Bucur and G. Iacca, “Influence maximization in social networks with genetic algorithms”, *European Conference on the Applications of Evolutionary Computation*, Springer, pp. 379–392, 2016.

[19] P. Krömer and J. Nowaková, “Guided genetic algorithm for the influence maximization problem”, *International Computing and Combinatorics Conference*, Springer, pp. 630–641, 2017.

[20] C.-W. Tsai, Y.-C. Yang, and M.-C. Chiang, “A genetic new greedy algorithm for influence maximization

in social network", *IEEE International Conference on Systems, Man, and Cybernetics*, pp. 2549–2554, 2015.

پنجمین کنفرانس ملی انفورماتیک ایران
پژوهشگاه دانشهای بنیادی، پردیس فرمانیه، تهران
۱۳ و ۱۴ دی ماه ۱۴۰۲

اقدامات امنیتی برای مقابله با تهدیدات امنیتی در برنامه‌های کاربردی اینترنت اشیا

صدیقه هدایتی[۱]، پیام محمودی نصر[۲]
[۱] دانشکده فنی و مهندسی، دانشگاه مازندران ، بابلسر،
hedayaty6741@gmail.com
[۲] دانشکده فنی و مهندسی، دانشگاه مازندران ، بابلسر،
p.mahmoudi@umz.ac.ir

چکیده

در دنیای امروزه اینترنت اشیا به عنوان یک تکنولوژی محبوب می‌باشد. دستگاه‌های فیزیکی بی‌شماری در سراسر جهان به اینترنت متصل هستند. حسگرها و دستگاه‌های اینترنت اشیا با برنامه‌ها و سرویس‌هایی که در فضای ابری هستند ارتباط برقرار می‌کنند. این دستگاه‌ها در معرض خطر دسترسی هکرها به فضای خصوصی افراد هستند. برای اجرای صحیح چنین محیطی باید به مسائل امنیتی در لایه‌های مختلف این فناوری توجه شود. در این مقاله حوزه‌های کاربردی اینترنت اشیا مطرح می‌شود و سپس کاربرد هر یک از این لایه‌ها و تهدیدات موجود در هر یک از این لایه‌ها بیان شده و برخی اقدامات امنیتی برای غلبه بر مشکلات امنیتی ذکر می‌شود. همچنین راهکارهایی برای مقابله با تهدیدات امنیتی در محیط اینترنت اشیا مورد بررسی قرار می‌گیرد.

کلمات کلیدی

اینترنت اشیا، تهدیدات امنیتی، اقدامات امنیتی، راه حل امنیتی

۱- مقدمه

در اینترنت اشیا[۱] (IoT) آینده‌ای مدنظر است که در آن اشیا و سبک زندگی با کمک میکروکنترلرها، فرستنده‌های دیجیتال و پشته‌های پروتکل مناسب طراحی خواهند شد. اینترنت اشیا با دستگاه‌های مختلفی مانند لوازم خانگی، دوربین‌های نظارتی و حسگرها، وسایل نقلیه و غیره در تعامل است [1]. اتصال دستگاه‌های فیزیکی به اینترنت به سرعت در حال افزایش است. بر اساس گزارش اخیر گارتنر، انتظار می‌رود تعداد اتصالات ماشین به ماشین تا سال ۲۰۲۴ به ۲۷ میلیارد برسد[2]. با افزایش استفاده از این دستگاه‌ها اطلاعات موجود در این شبکه در معرض خطر قرار دارند. هکرها می‌توانند به طور غیرقانونی به این شبکه‌ها نفوذ کرده و بدون شناسایی برای مدت طولانی‌تری به داده‌ها دسترسی داشته باشند. در صورتی که تنظیمات امنیتی توسط این افراد دور زده شود، بدون شناسایی، احراز هویت و مجوز، محیط این شبکه قابل دسترس خواهد بود[3]. در حال حاضر، محیط اینترنت اشیا با مسائل امنیتی متعددی مواجه است. در این مقاله رایج‌ترین تهدیدهای مرتبط با برنامه‌های اینترنت اشیا به همراه راه‌حل‌های پیشرفته معرفی می‌شود که می‌تواند تاثیر بسزایی در کاهش اثر این تهدیدات داشته باشد. در بخش دوم این مقاله به مرور کارهای پیشین پرداخته می‌شود. بخش سوم به بررسی استفاده از اینترنت اشیا در بخش‌های مختلف جامعه می‌پردازد. در بخش چهارم، لایه‌های معماری اینترنت اشیا معرفی شده و در بخش پنجم تهدیدات امنیتی در هر یک از این لایه‌ها مورد تحلیل قرار می‌گیرد. در بخش ششم، چند نمونه از اقدامات امنیتی برای محافظت در برابر حملات پیشنهاد شده است. در نهایت در بخش هفتم راهکارهایی که برای افزایش امنیت اینترنت اشیا موثر هستند معرفی می‌شود.

۲- کارهای مرتبط

نویسندگان[1] معماری اینترنت اشیا را توضیح داده و سپس تهدیدات اصلی مربوط به حریم خصوصی و امنیت برنامه‌های کاربردی محبوب مبتنی بر اینترنت اشیا شرح داده می‌شود. نویسندگان [2] به بررسی چالش‌های مرتبط با امنیت و منابع تهدید کننده در برنامه‌های کاربردی اینترنت اشیا می‌پردازند. پس از بحث در مورد مسائل امنیتی، فناوری‌های مختلف با تمرکز بر دستیابی به درجه بالایی از اعتماد در برنامه‌های کاربردی اینترنت اشیا پیشنهاد می‌شود. نویسندگان [3] در مورد فناوری و معماری لایه‌ای IoT بر ویژگی‌های کلیدی خانه‌های هوشمند، کشاورزی هوشمند، حمل‌ونقل هوشمند و مراقبت‌های بهداشتی هوشمند تمرکز دارند. سپس، تهدیدات امنیتی و آسیب پذیری‌های موجود در هر لایه توضیح داده می‌شود. همچنین طبقه‌بندی چالش‌های امنیتی مانند محرمانگی، یکپارچگی، حریم خصوصی، در دسترس بودن، احراز هویت، عدم انکار و مدیریت کلید به طور کامل بررسی می‌شود. نویسندگان[4] یک بحث جامع در مورد وضعیت فعلی اینترنت اشیا ارائه نمودند. تمرکز ویژه آن‌ها در زمینه تهدیدات حریم خصوصی و امنیت، سطح حمله، آسیب‌پذیری‌ها می‌باشد. در ادامه طبقه‌بندی تهدیدات، الزامات و چالش‌های کاربران اینترنت اشیا شناسایی شده و مورد بحث قرار گرفته تا نیازها و نگرانی‌های اصلی امنیت و حریم خصوصی کاربران برجسته شود. نویسندگان[5] یک بررسی جامع از جدیدترین راه حل‌های پیشنهادی امنیتی و حفظ حریم خصوصی در اینترنت اشیا ارائه نمودند. همچنین مزایایی که رویکردهای جدید مانند بلاک‌چین و شبکه‌های نرم افزار محور[۲] می‌توانند برای امنیت و حریم خصوصی در IoT از نظر انعطاف‌پذیری و مقیاس‌پذیری داشته باشند، مورد بررسی قرار گرفت.

۳- حوزه‌های کاربردی اینترنت اشیا

۳-۱- بیمارستان هوشمند

در حوزه اینترنت اشیا کنترل سلامت بیماران از راه دور با در نظر گرفتن منابع موجود و با ارائه خدمات بهتر و بیشتر به بیماران کمک شده است. این سرویس به علت کاهش مراجعه غیر حضوری بیماران به بیمارستان، موجب رفاه بیماران می‌شود. در این رویکرد پارامترهای مختلف سلامت مانند سطح قند خون، فشار خون و ضربان قلب توسط دستگاه‌های هوشمند پزشکی کنترل خواهد شد. این دستگاه‌ها برای انجام این وظایف به سرور اینترنت اشیا متصل می‌شوند [6]. اطمینان از محرمانه بودن و یکپارچگی داده‌های مبادله شده باید در نظر گرفته شود. علاوه بر این، محل نصب دستگاه‌های اینترنت اشیا، هویت بیماران و غیره باید مخفی بماند [7].

۳-۲- کشاورزی هوشمند

در کشاورزی هوشمند مواردی مانند کنترل رطوبت خاک، شرایط آب و هوایی، آبیاری در مناطق خشک و کنترل دما در نظر گرفته شده است. هوشمندی در صنعت کشاورزی به دستیابی به محصول بیشتر کمک کرده و موجب کاهش زیان‌های مالی کشاورزان می‌شود. به منظور جلوگیری از قارچ و سایر آلودگی‌های میکروبی، درجه حرارت و رطوبت در تولید غلات و سبزیجات باید کنترل شود. کنترل شرایط آب و هوایی همچنین می‌تواند به افزایش عملکرد و کیفیت سبزیجات و محصولات زراعی کمک کند. اگر چنین برنامه‌هایی به خطر بیفتند، ممکن است منجر به از بین رفتن محصولات کشاورزی شود[2].

۳-۳- حمل و نقل هوشمند

نسل بعدی سیستم حمل و نقل، سیستم‌های حمل و نقل هوشمند است که هدف آن اتصال افراد، جاده‌ها و وسایل نقلیه با کمک توسعه سیستم‌های تعبیه شده و فناوری‌های ارتباطی است. با اتصال و توزیع پردازنده‌های هوشمند در داخل وسایل نقلیه و همچنین از طریق زیرساخت‌ها، حمل و نقل می‌تواند ایمن‌تر، سبزتر و راحت‌تر شود[5]. حریم خصوصی رانندگان باید از ناظران غیرمجاز محافظت شود. این شبکه‌ها باید قابل دسترس بوده و در مقابل حملات سیگنال‌های پارازیت[۳] که هدف آنها مختل کردن ارتباط بین وسایل نقلیه است، هوشمند باشند[8].

۳-۴- تولیدات هوشمند

اینترنت اشیا از فناوری‌های جدیدی مانند ارتباطات ماشین به ماشین، شبکه‌های حسگر بی‌سیم ، فناوری‌های اتوماسیون وکلان داده برای ایجاد یک اکوسیستم صنعتی هوشمند استفاده می‌کند. ارائه بهره وری بهتر، کارایی، قابلیت اطمینان و کنترل بهتر محصولات نهایی یکی از اهداف اصلی اینترنت اشیا است. سیستم تولید باید بتواند در شرایط بحرانی به کار خود ادامه دهد. بنابراین، یکپارچگی اطلاعات رد و بدل شده بین دستگاه‌های IoT در سیستم صنعتی باید حفظ شود. فرآیند تولید در برابر حملات جاسوسی حساس و محرمانه است، بنابراین داده‌ها، کدها و پیکربندی‌های سیستم باید با استفاده از مکانیزم‌های رمزگذاری محافظت شوند. در این سیستم‌ها برخی از وظایف تولیدی به اشخاص ثالث برون سپاری شده، بنابراین احراز هویت این افراد ضروری است[7].

۳-۵- اتوماسیون خانگی

یکی از پرکاربردترین کاربردهای اینترنت اشیا، اتوماسیون خانگی است. برنامه‌های کاربردی کنترل از راه دور برای وسایل الکتریکی به منظور صرفه‌جویی در مصرف انرژی، سیستم‌های مستقر بر روی پنجره‌ها و درها برای شناسایی سارقان، نمونه‌ای از این کاربردها هستند[2]. استفاده از الگوریتم‌های امنیتی مبتنی بر منطق برای بهبود سطح امنیت در خانه‌ها پیشنهاد شده است[9].

٤- لایه‌های معماری اینترنت اشیا

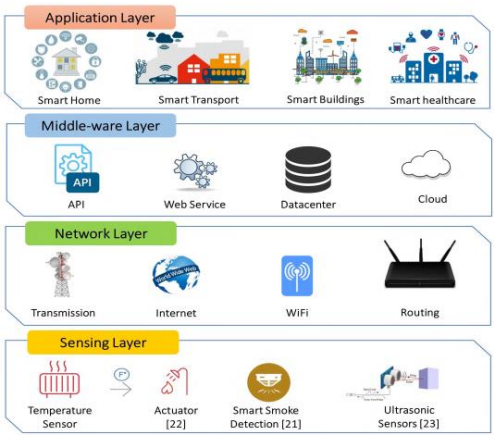
لایه‌های این معماری شامل لایه حس کننده، لایه شبکه، لایه میان افزار و لایه کاربرد است. داده‌ها از دو لایه پایین گرفته می‌شود و مسئولیت دو لایه بالا استفاده از داده‌ها در برنامه است. شکل یک لایه‌های این معماری را نشان می‌دهد[1].

٤-۱- لایه حس کننده

لایه حس کننده با محیط اطراف در تعامل است و هدف اصلی آن جمع‌آوری اطلاعات از محیط با استفاده از حسگرها می‌باشد. سپس اطلاعات به دست آمده برای پردازش بیشتر به لایه شبکه منتقل می‌شود [10, 11]. این لایه شامل قطعات سخت افزاری است، اجزایی مانند شبکه‌های حسگر بی‌سیم، رابط‌های داده الکترونیکی، سیستم‌های شناسایی فرکانس رادیویی[۴] (RFID)، سیستم‌های موقعیت یابی جهانی و غیره. در این لایه سنسورها هدف حمله قرار می‌گیرد [1].

٤-۲- لایه شبکه

لایه شبکه مانند یک سیستم عصبی مرکزی در سراسر شبکه عمل می‌کند. و نقش اصلی‌اش، مسیریابی و انتقال داده‌ها به مراکز و دستگاه‌های مختلف اینترنت اشیا از طریق اینترنت است[10]. این لایه نقش مهمی در انتقال ایمن و محرمانه اطلاعات حساس از لایه حس کننده به لایه‌های بالاتر دارد[11].



شکل (۱) : لایه‌های معماری اینترنت اشیا [2]

۳-٤- لایه‌ی میان‌افزار

در سیستم اینترنت اشیا انواع مختلفی از خدمات در طول ارتباط ارائه می‌شود. مدیریت خدمات بر عهده لایه میان افزار است و اطلاعات لایه پایین در پایگاه داده این لایه ذخیره می‌شود. علاوه بر این، قابلیت بازیابی، پردازش اطلاعات و سپس تصمیم‌گیری خودکار بر اساس نتایج محاسبات توسط این لایه صورت می‌گیرد[2].

٤-٤- لایه‌ی کاربرد

این لایه شامل تمام نرم افزارهای لازم برای ارائه یک سرویس خاص است. بر اساس اطلاعات تحلیل شده در لایه میان افزار، وظیفه مدیریت برنامه‌های جامع و تضمین یکپارچگی داده‌ها، محرمانه بودن داده‌ها و صحت داده‌ها بر عهده این لایه است. رابط برنامه با پروتکل‌های لایه پایین‌تر برای ارسال داده‌ها از طریق شبکه توسط پروتکل‌های لایه برنامه تعریف می‌شود. پروتکل‌های زیادی در این لایه در اینترنت اشیا وجود دارد که با توجه به نوع اپلیکیشن، پروتکل مناسب آن اپلیکیشن انتخاب و در شبکه مورد استفاده قرار می‌گیرد [11].

۵- تهدیدات امنیتی در لایه‌های معماری اینترنت اشیا

در بخش قبل در مورد معماری یک برنامه IoT بحث شد. اکنون برخی از موضوعات مربوط به تهدیدات رایج در هر یک از این لایه‌ها مورد بحث قرار می‌گیرد.

۵-۱- تهدیدات امنیتی لایه حس کننده

۵-۱-۱- ضبط سنسور

همانطور که گفته شد، برنامه‌های IoT از گره‌های کم مصرف مانند حسگرها و محرک‌ها ساخته شده‌اند. در صورتی که یک سنسور در یک سیستم اینترنت اشیا با یک گره مخرب جایگزین شود، سنسور جدید ممکن است بخشی از سیستم بوده اما توسط مهاجم کنترل می‌شود[2] .

۵-۱-۲- تزریق کد مخرب

در این حمله، کد مخرب توسط مهاجم در حافظه سنسور تزریق می‌شود. در نتیجه، ممکن است سنسورها مجبور به انجام برخی عملکردهای ناخواسته شوند[2].

۵-۱-۳- حمله راه‌اندازی[5]

اکثر مکانیسم‌های حفاظتی در طول فرآیند راه اندازی (بوت شدن) سیستم فعال نیستند و یک هکر می‌تواند در این مدت به بخش‌های حساس سخت افزار و نرم افزار دسترسی پیدا کند. دستگاه‌های کم مصرف دارای چرخه خواب و بیداری مداوم هستند که همین امر موجب آسیب‌پذیری بیشتر آنها در برابر این حملات می‌شود[1].

٤-۱-۵- استراق سمع و تداخل

اپلیکیشن‌های اینترنت اشیا معمولاً از حسگرهای مختلفی تشکیل می‌شوند که در محیط‌های باز قرار دارند. بنابراین، در حین انتقال داده یا احراز هویت، اطلاعات ممکن است توسط مهاجمان رهگیری و ثبت شود[2].

۵-۲- تهدیدات امنیتی لایه شبکه

۵-۲-۱- حمله فیشینگ[6]

هنگامی که کاربران از صفحات وب در اینترنت بازدید می‌کنند، امکان برخورد با سایت‌های فیشینگ وجود دارد. هنگامی که حساب کاربری و رمز عبور کاربر به خطر بیفتد، کل محیط اینترنت اشیا مورد استفاده کاربر در برابر حملات سایبری آسیب پذیر می‌شود[2].

۵-۲-۲- حمله انکار سرویس توزیع شده[7] (DDoS)

در این نوع حمله، سرورهای مورد نظر با تعداد زیادی درخواست ناخواسته پر می‌شود. با این کار سرور هدف قادر به پاسخگویی نیست و در نتیجه خدمات کاربران اصلی مختل می‌شود[2]. بسیاری از دستگاه‌ها در برنامه‌های IoT پیکربندی ضعیفی دارند، بنابراین به دروازه‌های آسانی برای مهاجمان تبدیل شده و در نتیجه حملات DDoS روی سرورهای هدف انجام می‌گیرد[12].

۵-۲-۳- حمله مسیریابی

در چنین حملاتی، مسیر داده‌ها توسط سنسورهای مخرب در حین انتقال داده تغییر می‌یابد. به عنوان مثال، در حملات گودال، جزئیات مسیریابی نادرست به سنسورهای متصل ارسال شده، بنابراین حجم زیادی از ترافیک شبکه بدین صورت جذب می‌شود. حمله از سنسوری سرچشمه می‌گیرد که توسط مهاجم در شبکه به خطر افتاده، بنابراین از این حمله می‌تواند برای راه اندازی انواع دیگر حملات استفاده کند[1]. حمله کرم چاله حمله دیگری است که اگر با حملات دیگری مانند حملات گودال ترکیب شود، به یک تهدید امنیتی شدید تبدیل می‌شود[2].

٤-۲-۵- جعل RFID

اطلاعات ارسال شده از طریق یک تگ RFID می‌تواند تغییر داده و جعل شود، زیرا مهاجم قادر به کپی سیگنال RFID است[1].

۵-۳- تهدیدات امنیتی لایه میان‌افزار

۵-۳-۱- حمله مرد میانی

این حمله نوعی استراق سمع است که هدف حمله، کانال ارتباطی است. تمام ارتباطات خصوصی بین دو نفر از طریق شخص غیر مجاز به طور مخفیانه کنترل می‌شود. حتی هویت قربانی می‌تواند توسط این شخص جعل شده و در ادامه اطلاعات بیشتری برای برقراری ارتباط عادی‌تر به دست آید[13].

۲-۳-۵- حمله داخلی

شناسایی این نوع حمله آسان نیست زیرا مهاجم می‌تواند یک عضو فعلی یا سابق با دسترسی واقعی به جزئیات و اطلاعات سیستم باشد و قادر به انجام انواع مختلف حملات باشد[1].

۳-۳-۵- حمله تزریق SQL[۸]

در این نوع حمله، کد SQL مخرب در یک برنامه جاسازی می‌شود. در نتیجه، مهاجم به اطلاعات خصوصی کاربر دسترسی پیدا کرده و حتی ممکن است رکوردهای موجود در پایگاه داده مورد تغییر قرار گیرد[2].

٤-۳-۵- تزریق بدافزار ابری

با تزریق بدافزار به فضای ابری، ماشین مجازی به فضای ابری تزریق می‌شود. در واقع، با ایجاد یک نمونه ماشین مجازی یا یک ماژول سرویس مخرب، وانمود می‌شود که یک سرویس معتبر است. به این ترتیب، مهاجم می‌تواند به درخواست‌های خدمات قربانی دسترسی داشته باشد[2]. پیچیدگی بیش از حد ناظر ارشد[۹]، تخصیص نامحدود منابع و انعطاف پذیری پیکربندی آن در فضای ابری این امکان فراهم شده تا از یک ماشین مجازی برای حمله به ماشین مجازی دیگر استفاده شود. در طول فرآیند مهاجرت یک ماشین مجازی اطلاعات غیرقانونی در دسترس مهاجمان قرار می‌گیرد[14].

٤-۵- تهدیدات امنیتی لایه‌ی کاربرد

۱-٤-۵- حمله‌ی بویش[۱۰]

در این نوع حمله، با استفاده از یک برنامه اسنیفر به سیستم صدمه وارد می‌شود. در صورتی که در حین انتقال، هیچ رمزگذاری روی بسته‌های داده وجود نداشته باشد، موجب دسترسی به اطلاعات شبکه توسط مهاجمان می‌شود[10] .

۲-٤-۵- حمله قطع سرویس

به این حملات، حملات وقفه غیرقانونی یا حملات DDoS نیز گفته می‌شود. موارد مختلفی از چنین حملاتی به برنامه‌های کاربردی اینترنت اشیا صورت گرفته است. چنین حملاتی با شلوغ کردن مصنوعی سرورها یا شبکه، موجب محرومیت کاربران قانونی از خدمات برنامه‌های کاربردی اینترنت اشیا می‌شوند[2, 10].

۳-٤-۵- حمله کنترل دسترسی

با مکانیزم کنترل دسترسی به کاربران یا فرآیندهای قانونی اجازه دسترسی به داده‌ها یا حساب داده می‌شود. حمله کنترل دسترسی یک حمله حیاتی در برنامه‌های اینترنت اشیا است زیرا هنگامی که دسترسی به خطر بیفتد، برنامه اینترنت اشیا به طور کامل در برابر حملات آسیب‌پذیر می‌شود[2].

٤-٤-۵- حمله برنامه‌ریزی مجدد

اگر برنامه نویسی در این لایه ناامن باشد، به راحتی هر دستگاه اینترنت اشیا از راه دور، توسط مهاجمین دوباره قابل برنامه‌ریزی است[1].

٦- اقدامات امنیتی

پس از بیان برخی حملات امنیتی مختلف اعمال شده توسط مهاجمان، در لایه‌های مختلف معماری اینترنت اشیا، در این بخش به برخی از الزامات امنیتی اشاره می‌شود.

۱-٦- احراز هویت

احراز هویت در هر لایه IoT مهم است. اطلاعات دریافت شده توسط خواننده باید از یک برچسب الکترونیکی معتبر ارسال شود. به عنوان مثال در لایه حسگر، گره‌های حسگر باید ابتدا احراز هویت شوند تا از حملات DDoS جلوگیری شود. اپن آیدی[۱۱] یک چارچوب استاندارد برای اهداف احراز هویت است. این روش یک استراتژی برای یک سایت فراهم کرده تا کاربر به مکان دیگری هدایت شود و با یک عبارت قابل تأیید برگردد.

۲-٦- تشخیص نفوذ

در تکنیک‌های تشخیص نفوذ، هرگونه فعالیت مشکوک با ایجاد یک هشدار در سیستم کنترل شده و با نظارت مستمر و فناوری‌های رایانش ابری و مجازی‌سازی، راه‌حل‌های زیادی برای تهدیدات ارائه می‌شود.

۳-٦- ارزیابی ریسک

این روش برای شناسایی تهدیدات در شبکه استفاده می‌شود و این فرآیند شامل تجزیه و تحلیل وضعیت، مقایسه استانداردهای مختلف و بررسی سطح پذیرش ریسک می‌باشد[10].

٤-٦- صدور گواهینامه و کنترل دسترسی

تایید هویت واقعی بین دو نهادی که به هم مرتبط هستند با صدور گواهینامه انجام می‌شود. با استفاده از زیرساخت کلید عمومی (PKI)، احراز هویت قوی با صدور گواهینامه‌های کلید عمومی دو طرفه برای شناسایی اصالت و محرمانه بودن سیستم اینترنت اشیا انجام می‌شود. برای اطمینان از امنیت، کنترل دسترسی با مسدود کردن دسترسی به ماشین‌ها، اشیاء یا افرادی که دسترسی غیرقانونی به منابع دارند، انجام می‌شود. برای کنترل دسترسی مؤثر، شناسایی صحیح نهاد باید با استفاده از فناوری صدور گواهینامه تضمین شود[5].

۵-٦- عدم انکار

عدم انکار یکی از جنبه‌های امنیتی است که تضمین می‌کند، اعضای ارتباط توانایی ارسال یا دریافت اطلاعات به صورت یکپارچه را دارند. علاوه بر این، این اطمینان ایجاد می‌شود که انتقال داده یا شناسایی بین دو شی IoT غیرقابل انکار است.

٦-٦- صحت داده‌ها

صحت داده به این معنی است که پیام توسط یک نهاد غیرمجاز در حین انتقال داده‌ها تغییر نیابد. بنابراین، تضمین می‌شود که گیرنده دقیقا همان چیزی را که منبع ارسال کرده، دریافت کرده است. هدف اصلی متوقف کردن یک نهاد

غیرمجاز است که سعی در ایجاد تغییرات غیرقانونی دارد. برای حفظ ایمنی دستگاه‌های هوشمند در شبکه IoT، باید یکپارچگی داده‌ها تضمین شود. علاوه بر این، مکانیزم‌های رمزنگاری و رمزگذاری زمانی که داده‌های ارسالی مهم هستند، باید اعمال شود[15].

۷- راه حل امنیتی

روش‌های مختلفی برای ایمن سازی محیط و برنامه‌های اینترنت اشیا وجود دارد که در اینجا به چند نمونه از آنها اشاره می‌شود:

۷-۱- راه حل‌های امنیتی در محاسبات مه[۱۲]

محاسبات مه، توسعه مدل رایانش ابری در لبه شبکه است. مه و ابر از منابع مشابه (شبکه، محاسبات و ذخیره سازی) استفاده نموده و از مکانیسم‌ها و ویژگی‌های مشابهی (مجازی سازی، چند اجاره‌ای) بهره برده اما به واسطه محاسبات مه مزایای بیشتری در سازماندهی و ایمن سازی داده‌ها ارائه می‌شود[16]. برخی از مزایای استفاده از محاسبات مه به شرح زیر است:

- اگر حمله‌ای به سیستم اینترنت اشیا رخ دهد، باید از لایه مه عبور کند که موجب شناسایی حمله شده و به این طریق اثرات مخرب آن کاهش می‌یابد.
- اطلاعات کاربران به جای کل شبکه با یک گره مه در ارتباط است، بنابراین امکان استراق سمع داده‌ها به دلیل کاهش ترافیک به حداقل می‌رسد.
- امکان ارائه داده‌ها در گره‌های مه به جای دستگاه‌های کاربران موجب می‌شود، خطر حملات به میزان قابل توجهی کاهش یابد[1].

۷-۲- یادگیری ماشین برای افزایش امنیت

یادگیری ماشین نشان داده که در محافظت از دستگاه‌های اینترنت اشیا در برابر حملات سایبری موثر است. در ادامه به برخی از این روش‌ها اشاره شده است:

- استفاده از یادگیری ماشینی باعث کاهش هزینه‌ها، بهبود رضایت مشتری و کاهش مصرف انرژی می‌شود[1].
- برای مقابله با حملات DDOS، یکی از روش‌ها استفاده از پرسپترون چند لایه[۱۳] (MLP) است که منجر به ایمنی شبکه در برابر این حملات می‌شود. به عنوان نمونه الگوریتم بهینه‌سازی ازدحام ذرات و الگوریتم پس انتشار برای آموزش MLP استفاده می‌شود.
- در روش‌های تشخیص حمله‌ای که از تخمین بردار حالت[۱۴] (SVE) استفاده شده، ابتدا وضعیت سیستم از روی اندازه‌گیری‌های مشاهده شده تخمین زده می‌شود. سپس باقی‌مانده‌ی اندازه‌گیری‌های مشاهده شده و برآورد شده به دست می‌آید. اگر باقی‌مانده از یک آستانه خاص فراتر رود، یک حمله تزریق داده شناسایی می‌شود.
- یکی از موثرترین تکنیک‌های مورد استفاده برای طراحی سیستم تشخیص نفوذ شبکه عصبی مصنوعی[۱۵] (ANN) است. هدف استفاده از ANN این است که بتواند از داده‌های ناقص تعمیم داده شده، داده‌های آنلاین را به عنوان نرمال یا مزاحم طبقه‌بندی نماید. در این شبکه امکان شناسایی و طبقه‌بندی فعالیت شبکه بر اساس منابع داده محدود، ناقص و غیرخطی فراهم شده و از ویژگی‌های محیط برای نشان دادن حضور هکر استفاده می‌شود [2].

۷-۳- فناوری بلاک‌چین[۱۶] برای بهبود امنیت

بلاک‌چین شامل یک پایگاه داده ایمن توزیع شده (یک دفتر کل عمومی) است که شامل تمام تراکنش‌های انجام شده توسط همه نهادهای شرکت کننده می‌باشد. هدف اصلی این پایگاه داده ارتباط گره‌های ناهمگن با یکدیگر به صورت کاملاً توزیع شده و ایمن بدون اتکا به هیچ نهاد مرکزی قابل اعتمادی است. اساساً هر گره در بلاک‌چین به گره دیگری اعتماد ندارد، زیرا این موجودیت به کل شبکه بلاک چین اعتماد دارد. هر گره دارای یک جفت کلید رمزنگاری (کلیدهای عمومی و خصوصی) است و از این طریق امکان تراکنش و تعامل با سایر گره‌های شبکه فراهم می‌شود. اولین راه حل مبتنی بر اینترنت اشیاء مبتنی بر بلاک‌چین توسط IBM توسعه یافت.

این پلتفرم ADEPT[۱۷] نامیده شده که شامل اثبات مفهوم یک پلت فرم غیرمتمرکز و ایمن اینترنت اشیا مبتنی بر پروتکل اتریوم است. بنابراین، نقش، مسئولیت و مجوزهای دستگاه‌های اینترنت اشیا به طور مستقل در کل اکوسیستم اینترنت اشیا تعریف و تنظیم می‌شود[7]. برخی از مزایای بلاک‌چین در امنیت اینترنت اشیا به شرح زیر است:

- با کمک معماری غیرمتمرکز بلاک‌چین راه‌حل‌های امنیتی مقیاس‌پذیرتر شده و مشکل شکست در یک نقطه خاص حل می‌شود. این فناوری در برابر حملات DDoS قدرتمندتر است.
- گره‌ها در بلاک‌چین با کلیدهای عمومی شناسایی می‌شوند. این نام‌های مستعار هیچ اطلاعاتی در مورد هویت گره‌های شرکت کننده نمی‌دهد.
- هر تراکنش قبل از ارسال به شبکه بلاک‌چین توسط یک گره امضا می‌شود و باید توسط ماینرها تایید شود. پس از تأیید، جعل یا تغییر تراکنش‌های ذخیره شده قبلی در بلاک چین غیرممکن است [5].

۷-۴- محاسبات لبه[۱۸] برای بهبود امنیت اینترنت اشیا

برای حل مشکلات رایانش ابری از محاسبات لبه به عنوان راه حل استفاده می‌شود. بدین صورت که یک سرور لبه کوچک بین کاربر و ابر یا مه قرار گرفته است. برخی از فعالیت‌های پردازشی به جای ابر در سرور لبه انجام می‌شود. در محاسبات لبه، توان محاسباتی و تحلیلی در لبه ارائه می‌شود. محاسبات لبه همچنین با اجتناب از نیاز به انتقال همه داده‌ها به ابر به کاهش هزینه‌های ارتباطی کمک می‌کند[17]. برخی از اثرات مثبت پیاده‌سازی محاسبات لبه در برنامه‌های کاربردی اینترنت اشیا عبارتند از:

- در محاسبات لبه، تمام داده‌ها در دستگاه یا شبکه محلی ذخیره و پردازش می‌شوند. هیچ حرکت داده‌ای از منبع داده به پردازنده وجود ندارد. این کار از انتقال داده‌ها جلوگیری و در نتیجه از خطر سرقت داده‌ها و دستکاری داده‌ها جلوگیری می‌شود.
- اکثر کشورها اقدامات نظارتی سختگیرانه‌ای برای جلوگیری از انتقال داده‌ها به خارج از مرزهای خود دارند، با استفاده از محاسبات لبه، داده‌ها در داخل مرزهای خود نگه داشته شده و از انطباق با قوانین حاکمیت داده اطمینان حاصل می‌شود.

- دستیابی به زمان پاسخگویی سریع یکی از مزایای محاسبات لبه است. اگر تأخیر در پاسخ‌ها وجود داشته باشد، ممکن است منجر به مشکلات ایمنی فیزیکی شود. به عنوان مثال، در دوربین مدار بسته تلویزیونی ناهنجاری‌ها با استفاده از محاسبات لبه تجزیه و تحلیل شده و داده‌های خلاصه و مشکوک با زمان پاسخ سریعتر به مراکز داده ارسال می‌شود[2 ,1].

۸- نتیجه‌گیری

در این مقاله به طور مختصر در مورد استفاده از اینترنت اشیا در بخش‌های مختلف زندگی امروز توضیحاتی داده شد و سپس لایه‌های معماری اینترنت اشیا و تهدیدات امنیتی که در هر یک از این لایه‌ها وجود دارد مورد بررسی قرار گرفت و نمونه‌هایی از اقدامات امنیتی برای غلبه بر مشکلات امنیتی ذکر شد. همچنین در مورد راهکارهای مقابله با تهدیدات امنیتی اینترنت اشیا از جمله محاسبات مه، یادگیری ماشین، فناوری بلاک‌چین و محاسبات لبه توضیحاتی ارائه شد. با وجود راهکارهای امنیتی مناسب هنوز نگرانی‌هایی در مورد اطمینان از امنیت مناسب در محیط اینترنت اشیا وجود دارد که نیازمند تحقیقات بیشتر در این زمینه است.

مراجع

1. Anand, S. and A. Sharma, *WITHDRAWN: Assessment of security threats on IoT based applications*. 2020, Elsevier.
2. Hassija, V., et al., *A survey on IoT security: application areas, security threats, and solution architectures.* IEEE Access, 2019. **7**: p. 82721-82743.
3. Khan, Y., et al., *Architectural Threats to Security and Privacy: A Challenge for Internet of Things (IoT) Applications.* Electronics, 2022. **12**(1): p. 88.
4. Ogonji, M.M., G. Okeyo, and J.M. Wafula, *A survey on privacy and security of Internet of Things.* Computer Science Review, 2020. **38**: p. 100312.
5. Kouicem, D.E., A. Bouabdallah, and H. Lakhlef, *Internet of things security: A top-down survey.* Computer Networks, 2018. **141**: p. 199-221.
6. Nazir, S., et al., *Internet of things for healthcare using effects of mobile computing: a systematic literature review.* Wireless Communications and Mobile Computing, 2019. **2019**: p. 1-20.
7. Kouicem, D.E., *Security of Internet of Things for systems of systems*. 2019, Université de Technologie de Compiègne.
8. Pirayesh, H. and H. Zeng, *Jamming attacks and anti-jamming strategies in wireless networks: A comprehensive survey.* IEEE communications surveys & tutorials, 2022. **24**(2): p. 767-809.
9. Jose, A.C. and R. Malekian, *Improving smart home security: Integrating logical sensing into smart home.* IEEE Sensors Journal, 2017. **17**(13): p. 4269-4286.
10. Swamy, S.N., D. Jadhav, and N. Kulkarni. *Security threats in the application layer in IOT applications*. in *2017 International conference on i-SMAC (iot in social, mobile, analytics and cloud)(i-SMAC)*. 2017. IEEE.
11. Lombardi, M., F. Pascale, and D. Santaniello, *Internet of things: A general overview between architectures, protocols and applications.* Information, 2021. **12**(2): p. 87.
12. Kolias, C., et al., *DDoS in the IoT: Mirai and other botnets.* Computer, 2017. **50**(7): p. 80-84.
13. Wong, H. and T. Luo. *Man-in-the-middle attacks on mqtt-based iot using bert based adversarial message generation*. in *KDD 2020 AIoT Workshop*. 2020.
14. Abdul-Ghani, H.A., D. Konstantas, and M. Mahyoub, *A comprehensive IoT attacks survey based on a building-blocked reference model.* International Journal of Advanced Computer Science and Applications, 2018. **9**(3): p. 355-373.
15. Azrour, M., et al., *Internet of things security: challenges and key issues.* Security and Communication Networks, 2021. **2021**: p. 1-11.
16. Atlam, H.F., R.J. Walters, and G.B. Wills, *Fog computing and the internet of things: A review.* big data and cognitive computing, 2018. **2**(2): p. 10.
17. Yu, W., et al., *A survey on the edge computing for the Internet of Things.* IEEE access, 2017. **6**: p. 6900-6919.

---

۱ Internet of thing
۲ Software define networks
۳ Jamming signal
۴ Radio frequency identification
۵ Boot attack
۶ Phishing attack
۷ Distributed denial of service
۸ Structured query language
۹ Hypervisor
۱۰ Sniffing attack
۱۱ Open id
۱۲ Fog computing
۱۳ Multilayer perceptron
۱۴ State vector estimator
۱۵ Artificial neural network
۱۶ Blockchain
۱۷ Autonomous decentralized Peer-to-Peer telemetry
۱۸ Edge computing

پنجمین کنفرانس ملی انفورماتیک ایران
پژوهشگاه دانشهای بنیادی، پردیس فرمانیه، تهران
۱۳ و ۱۴ دی ماه ۱۴۰۲

طبقه بندی سیگنال های قلبی توسط شبکه های عصبی convolutional و SqueezeNet

سیده محبوبه مولوی عربشاهی[1]، فاطمه معاون[2]

[1] استادیار، گروه ریاضی کاربردی، دانشکده ریاضی و علوم کامپیوتر، دانشگاه علم و صنعت ایران، تهران
molavi@iust.ac.ir

[2] دانشجو ارشد ریاضی کاربردی، دانشکده ریاضی و علوم کامپیوتر، دانشگاه علم و صنعت ایران، تهران، moaven_fatemeh79@mathdep.iust.ac.ir

چکیده

در این تحقیق، از شبکه عصبی کانولوشنال و شبکه عصبی SqueezeNet برای طبقه‌بندی سیگنال‌های قلبی استفاده شده است. از ۱۶۲ ضبط ECG استفاده شده است. این ضبط‌ها شامل ۹۶ ضبط از افراد مبتلا به آریتمی، ۳۰ ضبط از افراد مبتلا به نارسایی احتقانی قلب و ۳۶ ضبط از افراد با ریتم طبیعی سینوسی است. هدف این تحقیق، تمایز بین ARR (آریتمی)، CHF (نارسایی احتقانی قلب) و NSR (ریتم طبیعی سینوسی) است. تحقیق حاکی از اهمیت استفاده از تحلیل سیگنال ضربان قلب برای تشخیص زودهنگام بیماری‌های قلبی است. این روش می‌تواند در تشخیص و درمان به موقع این بیماری‌ها مفید باشد و از خطرات و امکان عدم دقت تشخیص توسط پزشکان کاسته شود.

نمونه‌ها به عنوان ورودی به شبکه‌های عصبی داده شده‌اند و نتایج این دو شبکه عصبی با یکدیگر مقایسه شده است. نتایج نشان می‌دهد که شبکه عصبی SqueezeNet دقت ۹۵.۶۵٪ و شبکه عصبی کانولوشنال دقت ۹۲.۵۰٪ در طبقه‌بندی سیگنال‌های قلبی دارند.

کلمات کلیدی

سیگنال الکتروکاردیوگرافی(ECG)، داده کاوی، شبکه عصبی، طبقه بندی، یادگیری عمیق، آریتمی قلبی(ARR)، نارسایی احتقانی قلب (CHF)، ریتم سینوسی طبیعی(NSR)

1- مقدمه

آریتمی، یک ضربان قلب غیرطبیعی است که می‌تواند تهدیدکننده زندگی افراد باشد. در واقع به هر گونه تغییر از توالی‌های طبیعی تکانه‌های الکتریکی که باعث آهنگ غیرطبیعی قلب می‌شود، گفته می‌شود. در سال‌های اخیر، بیماری‌های قلبی–عروقی و تعداد مرگ و میر ناشی از آن به طور قابل توجهی افزایش یافته است. قلب به عنوان یک عضو مهم و نقشی اساسی در عملکرد بدن انسان ایفا می‌کند. بیماری‌های قلبی دیگر نیز از جمله سکته، نارسایی قلبی، به طور معمول با آریتمی همراه هستند.

بیماری قلبی یک اصطلاح کلی است و شامل تمام بیماری‌هایی است که بر اثر عوامل مختلف بر روی قلب تأثیر می‌گذارند. بیماری کرونر قلب، بیماری التهاب عضله قلب، آنژین صدری و... جزء بیماری‌های قلبی و عروقی محسوب می‌شوند. تشخیص درست و به موقع آریتمی‌های قلبی از اهمیت بسیاری برخوردار است. یکی از راه‌های شناخته شده برای تشخیص به موقع این آریتمی‌ها، بررسی فعالیت‌های الکتریکی قلب با استفاده از سیگنال‌های الکتروکاردیوگرافی (ECG) است. تغییرات معنی‌داری از ساختار قلب بیماران و ضربان‌های قلب آنها با استفاده از این سیگنال‌ها قابل تشخیص هستند. به تبع آن، انجام فرآیند تشخیص الگو جهت تشخیص ویژگی و طبقه‌بندی سیگنال ECG، قابل توجه می‌باشد.

بررسی دقیق نارسایی‌های قلبی مستلزم تهیه نمونه‌های متعدد و بررسی آنها توسط پزشک ماهر است. این نیز به نوبه خود وقت‌گیر بوده و نیازمند صرف زمان زیادی از سوی متخصص می‌باشد، از این رو احتمال خطا در هنگام تشخیص نیز افزایش می‌یابد.

در چندین سال اخیر، طبقه‌بندی خودکار سیگنال‌های ECG توجه زیادی را به خود جلب کرده است. این سیگنال‌ها به متخصص قلبی برای تحلیل و تشخیص آسان‌تر بیماری‌های قلبی می‌دهد. الگوریتم‌های هوش مصنوعی و یادگیری ماشین می‌توانند در تشخیص آریتمی‌ها و بیماری‌های قلبی به کمک پزشکان باشند.

به طور کلی، الگوریتم‌های یادگیری ماشینی می‌توانند با تحلیل دقیق سیگنال‌های ECG، الگوهای طبیعی و غیرطبیعی آریتمی‌ها را شناسایی کنند. این الگوریتم‌ها می‌توانند با استفاده از داده‌های طیف وسیعی از سیگنال‌های ECG که قبلاً توسط پزشکان بررسی و برچسب‌گذاری شده‌اند، آموزش ببینند. سپس با استفاده از این آموزش، می‌توانند سیگنال‌های جدید را تحلیل کرده و آریتمی‌ها و بیماری‌های قلبی را شناسایی کنند.

با این روش، می‌توان زمان و هزینه تشخیص بیماری‌های قلبی را کاهش داد و دقت تشخیص را افزایش داد. با این حال، باید توجه داشت که تشخیص نهایی و درمان همچنان به عهده پزشکان و متخصصان قلب می‌باشد و الگوریتم‌های یادگیری ماشینی فقط به عنوان یک ابزار کمکی برای تشخیص اولیه و فیلتر کردن موارد مشکوک استفاده می‌شوند.

در این پژوهش، یک الگوریتم جدید و مؤثر برای طبقه‌بندی سیگنال‌های قلبی با استفاده از دو شبکه عصبی کانولوشنال و SqueezeNet معرفی شده است. در این پژوهش به طبقه‌بندی سه شکل مختلف سیگنال ECG که دارای بیشترین اهمیت هستند، توجه شده است. این شکل‌ها شامل موارد زیر می‌شوند:

1) افراد مبتلا به آریتمی قلبی (ARR)
2) افراد مبتلا به نارسایی احتقانی قلب (CHF)

3) افراد با ریتم سینوسی طبیعی (NSR)

شبکه عصبی SqueezeNet ساختاری پیشرفته‌تر و جامع‌تر از شبکه عصبی کانولوشنال است. بنابراین، مقایسه توانایی این دو شبکه با یکدیگر می‌تواند به لحاظ ارزشمندی در نظر گرفته شود. در تحقیقات انجام شده تاکنون، به مقایسه این دو شبکه در طبقه‌بندی سیگنال‌ها پرداخته نشده است. در این مقاله، ما به این کار می‌پردازیم و مقایسه‌ای بین این دو شبکه در طبقه‌بندی سیگنال‌ها ارائه می‌دهیم.

2- مواد و روش ها

2-1- پایگاه داده

یکی از عوامل مهم در تحقیقات علمی و آزمایشگاهی، انتخاب پایگاه داده مناسب است. سیگنال ECG یک خروجی الکتریکی است که توسط قلب تولید می‌شود و توسط الکترودهای متصل به قفسه سینه شناسایی می‌شود. پایگاه داده ECG یک پایگاه داده است که حاوی سیگنال‌های الکتروکاردیوگرافی (ECG) است. پایگاه‌های داده برای تجزیه و تحلیل و بررسی اختلالات قلبی عروقی، تشخیص بیماری‌های قلبی و مطالعه ویژگی‌های فیزیولوژیکی قلب استفاده می‌شود. منابعی که پایگاه داده ECG را تشکیل می‌دهند، ممکن است شامل سیگنال‌های ECG از افراد سالم و بیمار باشد.

پایگاه داده مورد استفاده در این مقاله، پایگاه داده PhysioNet است که یکی از بزرگترین و معروف‌ترین پایگاه‌های داده مرتبط با سیگنال‌های فیزیولوژیکی است. این پایگاه داده شامل بیش از 30 پایگاه داده مختلف از جمله پایگاه‌های داده مربوط به سیگنال‌های ECG، سیگنال‌های EEG، سیگنال‌های EMG، سیگنال‌های فشار خون، سیگنال‌های تنفسی و سایر سیگنال‌های بدن است. همچنین، این پایگاه داده از فرمت‌های مختلف برای سیگنال‌ها از جمله فرمت‌های WFDB، EDF و MATLAB پشتیبانی می‌کند.

داده‌های مورد استفاده در این تحقیق در قالب mat می‌باشد.

2-2- مطالعه اول

همانطور که در بخش‌های قبلی بیان شد، هدف این تحقیق مقایسه دقت دو شبکه عصبی در طبقه‌بندی سیگنال‌های ECG است. بنابراین، در ابتدا سیگنال‌ها به شبکه عصبی کانولوشنال به منظور طبقه‌بندی داده شدند.

2-2-1- شبکه عصبی کانولوشنال

شبکه‌های عصبی کانولوشنال (CNN) برای شبیه‌سازی فعالیت‌های عصبی در مغز طراحی شده‌اند و معمولاً به عنوان سیستمی از واحدهای پردازشی به هم پیوسته نشان داده می‌شوند که به عنوان نورون‌های مصنوعی شناخته می‌شوند. این نورون‌ها داده‌های ورودی را دریافت می‌کنند، محاسبات را انجام می‌دهند و خروجی‌های مربوطه را تولید می‌کنند. یک نورون مصنوعی معمولاً شامل یک وزن (wi) است که نشان دهنده قدرت ارتباط بین نورون‌ها است، یک متغیر ورودی (xi) و یک بردار آستانه (b) که برای تعیین اینکه آیا نورون خروجی تولید می‌کند یا خیر، به کار می‌رود، تشکیل شده است.

$$u = \sum_{i}^{} w_i x_i \qquad (1)$$

$$z = f(u+b) = f\left(\sum_{i} w_i x_i + b\right) \qquad (2)$$

u، z، x، w و b به ترتیب کل ورودی، خروجی، متغیرهای ورودی، وزن‌ها و بایاس را نشان می‌دهند. تابع f نشان دهنده یک تابع فعال سازی است. مانند یک تابع خطی، سیگموئید، هذلولی یا صاف.

یک تابع فعال‌سازی خطی اصلاح‌شده به نام واحد خطی اصلاح‌شده (ReLU) در این مطالعه به عنوان تابع فعال‌سازی استفاده شد. معادله 3 تعریف تابع ReLU را ارائه می دهد [34].

$$f(u) = \max(0,u) = f\begin{cases} u(u>0) \\ u(u\le 0) \end{cases} \qquad (3)$$

در نهایت، پس از تکمیل همه لایه‌های کانولوشنال، ترکیبی از لایه‌های کاملاً متصل و طبقه‌بندی‌کننده معمولاً برای اختصاص یک برچسب که کلاس پیش‌بینی‌شده یک تصویر را نشان می‌دهد، استفاده می‌شود. در این تحقیق از تابع SoftMax برای تخمین احتمال یک تصویر ورودی متعلق به هر کلاس کاندید استفاده شد. این از نظر ریاضی به صورت معادله 4 نشان داده می شود:

$$y_k = softmsx_k(u_1,u_2,\dots,u_k) = \frac{e^{u_k}}{\sum_{i=1}^{k} e^{u_j}} \qquad (4)$$

در اینجا، 'k' مقدار عناصر خروجی را نشان می دهد، و 'u' نشان دهنده متغیرهای مستقل است. به منظور ارزیابی اثربخشی شبکه، لازم است یک تابع ضرر ایجاد شود. تابع ضرر اندازه گیری می کند که شبکه تا چه حد می تواند الگوهای موجود در مجموعه داده آموزشی را ضبط کند. هدف آموزش به حداقل رساندن اختلاف بین مقادیر پیش بینی شده و واقعی تولید شده توسط تابع ضرر است. معادله 5 تعریفی را برای اندازه گیری عدم تشابه بین توزیع های احتمال پیش بینی شده و واقعی تولید شده توسط تابع SoftMax ارائه می کند.

$$E = -\sum_{n=1}^{N}\sum_{k=1}^{K} t_{n,k} \log y_k \qquad (5)$$

در این جا، 't' مخفف بردار داده های آموزشی است، 'K' برای نشان دادن کلاس احتمال، و 'N' تعداد کل نمونه ها را نشان می دهد.

2-3- مطالعه دوم

در مطالعه دوم، سیگنال‌های ECG را به عنوان ورودی به شبکه SqueezeNet می‌دهیم.

2-3-1- شبکه SqueezeNet

مشکل اصلی شبکه‌های عصبی کانولوشن، تعداد زیاد پارامترهای آن است که مصرف حافظه را افزایش می‌دهد. استراتژی‌هایی برای کاهش تعداد پارامترها و بهبود دقت و در نتیجه اندازه شبکه کمتر از یک مگابایت استفاده می‌شود. یک لایه به نام squeeze برای کاهش تعداد کانال‌های ورودی با استفاده از فیلتر 3*3 مورد استفاده قرار می‌گیرد. این شبکه در سال 2016 توسط دانشمندان دانشگاه کالیفرنیا برای برنامه‌هایی که به دلیل محدودیت‌های حافظه و پردازش به شبکه‌های کوچک و سریع نیاز دارند، طراحی شد.

به طور معمول، چندین معماری CNN وجود دارد که می‌توانند به همان سطح دقت دست یابند. مدل‌های CNN با پارامترهای کمتر حداقل سه مزیت را ارائه می‌دهند که به همان میزان دقت مدل‌های بزرگ‌تر دست می‌یابند:

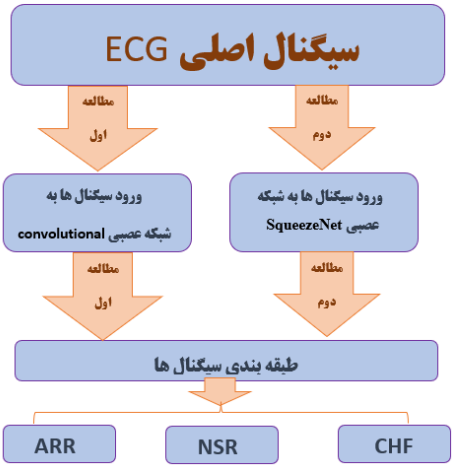
1. مدل‌های CNN با پارامترهای کمتر نیاز به ارتباط کمتری بین سرورها در زمینه آموزش با استفاده از دستگاه‌ها یا ماشین‌های متعدد دارند.

(2) مدل‌های CNN با پارامترهای کمتر، منابع شبکه کمتری را هنگام معرفی یک مدل جدید مصرف می‌کنند.

(3) مدل‌های CNN با پارامترهای کمتر، برای استقرار در FPGA و سایر سخت‌افزارهای با حافظه محدود، مناسب‌تر هستند.

SqueezeNet کاندیدای خوبی برای معماری CNN خواهد بود، به ویژه برای کاربردهایی که اندازه مدل کوچک مهم است. این معماری جدید CNN از طریق کاوش گسترده در فضای معماری‌های ممکن برای CNNها، در میان سایر معماری‌های جدید، شناسایی شد.

2-4- الگوریتم کلی از مطالعه اول و دوم

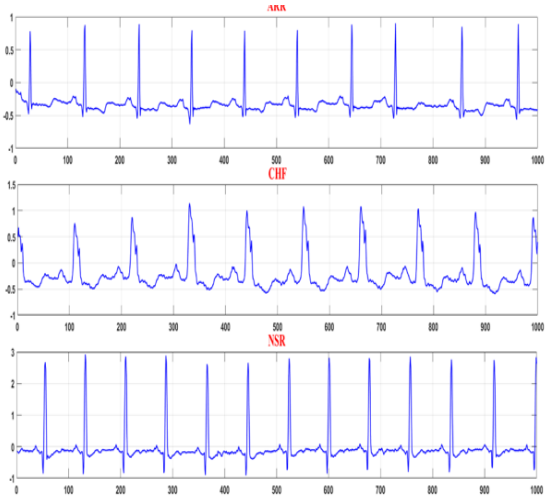


شکل (1): الگوریتم کلی فرآیند کار این تحقیق

3- نتایج

3-1- نمایش سیگنال های ECG

به منظور درک بهتر تفاوت بین این سه گروه از سیگنال‌های ECG، ابتدا یک نماینده از هر دسته ECG بر اساس نمایش ولتاژ–زمان ترسیم کردیم.



شکل (2): نمایش ECG هر دسته از طبقه بندی داده ها

3-2- طبقه بندی سیگنال ها

داده‌ها را به شبکه عصبی SqueezeNet تغذیه کردیم. داده‌های ورودی و شرایط شبکه عصبی، مانند حجم نمونه، تعداد داده‌های آموزشی، فرکانس آموزش و غیره، برای هر دو شبکه یکسان در نظر گرفته می‌شود. حجم داده‌ها در هر دسته به طور تصادفی به دو بخش آموزش و اعتبارسنجی تقسیم می‌شود. در این تحقیق، 60 درصد از تصاویر به عنوان داده‌های آموزشی و 40 درصد از آن‌ها به عنوان داده‌های اعتبارسنجی استفاده شده است.

سپس چند لایه اضافی به شبکه عصبی اضافه می‌شود تا برای طبقه‌بندی تصویر آماده شود. ابتدا یک لایه با ضریب 0.6 به شبکه اضافه می‌شود. این لایه برای جلوگیری از پیش‌پردازش در شبکه استفاده می‌شود. این کار به گونه‌ای است که در هر مرحله، تعدادی از ویژگی‌های ورودی به طور تصادفی حذف می‌شوند که به جلوگیری از برازش بیش از حد کمک می‌کند.

سپس یک لایه یادگیری با تعداد خروجی برابر با تعداد رده‌های موجود در داده‌های آموزشی به شبکه اضافه می‌شود. از ورودی این لایه، خروجی لایه قبلی و خروجی آن به عنوان خروجی شبکه استفاده می‌شود.

در نهایت با افزودن یک لایه طبقه‌بندی به شبکه، فرآیند طبقه‌بندی تصاویر انجام می‌شود. خروجی این لایه حاوی برچسب نهایی تصویر است.

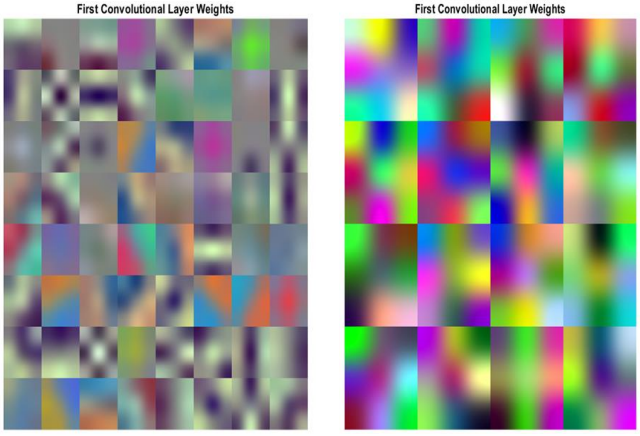
در شبکه‌های عصبی، وزن‌ها یا پارامترهای شبکه عمدتاً به عنوان ماتریس اعداد در هر لایه شبکه تعریف می‌شوند. این وزن‌ها با توجه به خطاهای مشاهده شده در خروجی شبکه ورودی به طور خودکار در فرآیند آموزش شبکه تنظیم می‌شوند تا شبکه بتواند بهترین عملکرد ممکن را در پاسخگویی مجدد به ورودی‌های جدید ارائه دهد.

وزن لایه‌های مختلف در شبکه به دلیل کاربردهای متفاوت، اهمیت و نقش متفاوتی در عملکرد شبکه دارند. با توجه به نوع شبکه و مشکل مورد نظر، بهینه‌سازی وزن لایه‌های مختلف در شبکه متفاوت است. به عنوان مثال، در شبکه‌های عصبی کانولوشن، وزن‌های لایه اول به دلیل اعمال عمل کانولوشن بر روی تصویر ورودی، می‌توانند ویژگی‌های متفاوتی را از تصویر استخراج کنند. در حالی که در شبکه‌های عصبی مکرر، وزن لایه‌های پنهان در تخمین مقادیر بعدی یا زمانی داده‌های ورودی مهم است.

وزن لایه اول در شبکه‌های کانولوشن به دلیل اعمال عملیات کانولوشن بر روی تصویر ورودی، می‌تواند ویژگی‌های متفاوتی را از تصویر استخراج کند. به عنوان مثال، فیلترهای کانولوشنال در وزن لایه اول می‌توانند لبه‌ها، خطوط و دیگر الگوهای ساده را در تصویر ورودی تشخیص دهند. با استفاده از این ویژگی‌های استخراج شده، شبکه می‌تواند الگوهای پیچیده‌تری را در تصویر تشخیص دهد. علاوه بر این، وزن لایه اول می‌تواند تا حد زیادی بر دقت شبکه تأثیر بگذارد. از آنجایی که وزن لایه اول نزدیک به تصویر ورودی است و مستقیماً بر ورودی‌های شبکه تأثیر می‌گذارد، طراحی وزن لایه اول تأثیر زیادی در عملکرد شبکه دارد. به همین دلیل، طراحی وزن لایه اول با توجه به ویژگی‌های مختلف تصویر ورودی و هدف شبکه می‌تواند دقت شبکه را به میزان قابل توجهی بهبود بخشد.

در شبکه SqueezeNet، به دلیل کاهش اندازه مدل، وزن لایه اول بسیار کمتر از وزن لایه اول در شبکه‌های دیگر است. این باعث می‌شود حجم داده‌های ورودی بسیار کمتر شود و سرعت پردازش افزایش یابد. همچنین، وزن لایه اول در SqueezeNet از روش squeeze-and-excitation استفاده می‌کند که باعث بهبود قابل توجهی در دقت شبکه می‌شود. در این روش، وزن‌های لایه اول با استفاده از لایه فشاری از فضای نگاشت پیچیدگی‌های قبلی استخراج می‌شوند و سپس این وزن‌ها با استفاده از یک لایه تحریک ترکیب می‌شوند تا وزن‌های بهتری به دست آید. این رویکرد به منظور افزایش وزن لایه اولیه و افزایش دقت شبکه استفاده می‌شود.

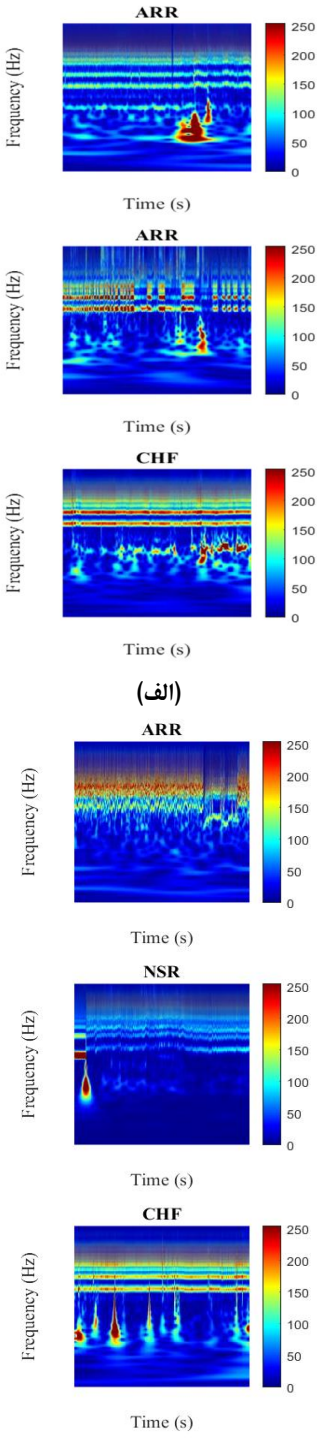
در ادامه می‌توانید نمایش تصویری از وزن‌های لایه اول شبکه‌های عصبی را مشاهده کنید:



(الف) **(ب)**

شکل (3): الف)شکل مقادیر وزن لایه اول در شبکه کانولوشن، که نشان دهنده فیلترهای آموخته شده برای استخراج ویژگی در وظایف پردازش تصویر است. ب) شکل وزن لایه اول در شبکه SqueezeNet

در شکل های زیر 3 تصویر تصادفی از تصاویر طبقه بندی شده توسط شبکه های عصبی را مشاهده می کنید:



(الف) **(ب)**

شکل (4): (الف) نمایش تصاویر تصادفی طبقه بندی شده توسط شبکه عصبی کانولوشن (ب) ارائه تصاویر تصادفی از تصاویر طبقه بندی شده توسط SqueezeNet

4- نتیجه گیری

با مقایسه خروجی تولید شده توسط شبکه‌های عصبی با برچسب‌های حقیقت، می‌توانیم صحت آنها را تعیین کنیم. دقت شبکه عصبی SqueezeNet در طبقه‌بندی داده‌ها 95.65 درصد و دقت شبکه عصبی کانولوشن در طبقه‌بندی همان داده‌های ورودی 92.50 درصد بود.

ویژگی‌ها و تنظیمات شبکه‌های عصبی، از جمله مقدار داده‌های ورودی، تعداد نقاط داده، میزان داده‌های آموزشی، و مدت زمان فرآیند آموزش و غیره، برای هر دو شبکه یکسان در نظر گرفته شد و البته درصد دقت شبکه‌ها در انجام این کار متفاوت بود که نشان می‌دهد شبکه عصبی SqueezeNet عملکرد بهتر و دقیق‌تری در طبقه‌بندی سیگنال‌های ECG دارد.

به طور کلی، نوآوری این تحقیق نسبت به تحقیقات قبلی در استفاده از مدل یادگیری عمیق SqueezeNet قابل مشاهده است. تا کنون توانایی و دقت بالای این مدل در طبقه‌بندی سیگنال‌های قلبی در تحقیقات قبلی ذکر نشده بود. همچنین، طبق این مدل یادگیری عمیق، SqueezeNet یک مدل بهبود یافته از مدل یادگیری کانولوشنی است. بنابراین، مقایسه این دو مدل یادگیری در شرایط یکسان بسیار مهم است. با توجه به نتایج، مشاهده شد که مدل یادگیری عمیق SqueezeNet در مقایسه با مدل یادگیری کانولوشنی از دقت بیشتری برخوردار است. ۱

روش‌های مختلفی برای دسته‌بندی سیگنال‌های قلبی ارائه شده است و روش پیشنهادی در این تحقیق در مقایسه با مطالعات انجام شده در این زمینه بهبود یافته است. الگوریتم یادگیری عمیق یک روش جدید است و هنوز پتانسیل زیادی برای بهبود و توسعه دارد. روش مورد استفاده در این تحقیق روشی نوآورانه در طبقه‌بندی سیگنال‌های قلبی است و نیاز به افزایش درصد دقت دارد. چندین مورد را می‌توان برای افزایش کارایی الگرویتم شبکه در کارهای آینده در نظر گرفت؛ مانند اصلاح تابع هزینه، تغییر تابع فعّال‌سازی، تنظیم تعداد داده‌های ورودی، بهبود روش شناسی آموزش، به‌روزرسانی داده‌های آموزشی و غیره. هر یک از این تغییرات می‌تواند در روند آموزش شبکه تأثیر گذار باشد و در نهایت باعث افزایش دقت شبکه گردد.

مراجع

[1] ابراهیم زاده، الیاس و پویان، محمد، **"پیش بینی مرگ ناگهانی قلبی (SCD)با استفاده از تحلیل های زمان -فرکانس سیگنال الکتروکاردیوگرام"**، مجله سیستم های هوشمند در مهندسی برق، سال سوم، شماره 4، صفحه 15-26، 1391

[2] پورآهنگریان، فرشته، کیانی، آزاده، کرمی، علی و زنج، بهمن، **"طراحی یک سیستم هوشمند مبتنی بر شبکه های عصبی و ویولت رای تشخیص آریتمی های قلبی"**، مجله مهندسی برق و الکترونیک ایران، سال نهم، شماره اول، 1391

[3] فولادی، صابر، فرسی، حسن و فرسی، فریما، **"به کارگیری فیلتر وفقی برای حذف نویز از سیگنالهای ECGبا استفاده از تبدیل موجک و یادگیری عمیق"**، مجله انفورماتیک سلامت و زیست پزشکی مرکز تحقیقات انفورماتیک پزشکی، دوره هفتم، شماره 3، صفحه 318-325، 1399

[4] قیومی زاده، حسین، فیاضی، علی، رضایی، خسرو، قلی زاده، محمد حسین و اسکندری، مهدی، **"جداسازی ناحیه گوشک دهلیز چپ در تصاویر اکوکاردیوگرافی قلبی با استفاده از شبکه عصبی عمیق"**، مجله انفورماتیک سلامت و زیست پزشکی مرکز تحقیقات انفورماتیک پزشکی، دوره هشتم، شماره 3، صفحه 315-325، 1400

[5] کاظمی، مریم، مهدی زاده، حسین و شیری، اردشیر، **"پیش بینی بیماری قلبی با استفاده از تکنیک داده کاوی شبکه عصبی"**، مجله علمی دانشگاه علوم پزشکی ایلام دوره بیست و پنج، شماره 1، صفحه 30-42، 1396

[6] محمدپور تهمتن، رضاعلی، اسماعیلی، محمدهادی، قائمیان، علی و اسمعیلی، جواد، **"کاربرد شبکه عصبی مصنوعی جهت ارزیابی بیماری عروق کرونری قلب"**، مجله دانشگاه علوم پزشکی مازندران دوره بیست و یکم شماره 86، صفحه 9-17، 1390

[7] Acharya U, Fujita R, Lih O, Hagiwara Y, Tan J, Adam M, Automated Detection of Arrhythmias Using Different Intervals of Tachycardia ECG Segments with Convolutional Neural Network. Information Sciences 405, 81-90, 2017.

[8] Alfaqui A, Daqrouq K, ECG signal denoising by wavelet transform thresholding. applied sciences 5, 276-281,2008.

[9] Barni M, Failla P, Lazzeretti R, Sadeghi A, Schneider R, TPrivacypreserving ECG classification with branching programs and neural networks. IEEE Transactions on Information Forensics and Security 602, 452-468, 2011.

[10] Bhatia S, Pandey S, Kumar A, Alshuhail A (2022) Classification of Electrocardiogram Signals Based on Hybrid Deep Learning Models. Sustainability. https://doi.org/10.3390/su142416572.

[11] Cai H, Xu L, Xu J, Xiong Z, Zhu C, Electrocardiogram Signal Classification Based on Mix Time-Series Imaging, Electronics,2022, https://doi.org/10.3390/electronics11131991.

[12] Carreiras A, Lourenço H, Silva A, Fred A, unifying approach to ECG biometric recognition using the wavelet transform. in International Conference Image Analysis and Recognition 7950, 53-62, 2013.

[13] Chang S, Tseng Y, Chen J, Chiu F, Tsai C, Hwang J, Wang Y and Tsai C, an artificial intelligence-enabled ECG algorithm for identifying ventricular premature contraction during sinus rhythm. European Journal of Medical Research,2022, https://doi.org/10.1186/s40001-022-00929-z.

[14] Dokur Z, Olmez T, Yazgan E, Comparison of Discrete Wavelet and Fourier Transforms for ECG Beat Classification. Electronic Letters, 2005, https://doi.org/10.1049/el:19991095.

[15] Fu L, Lu B, Nie B, Peng Z, Liu H, Pi X, Hybrid Network with Attention Mechanism for Detection and Location of Myocardial Infarction Based on 12-Lead Electrocardiogram Signals, Sensors,2020, https://doi.org/10.3390/s20041020.

[16] Geng Q, Liu H, Gao T, Liu R, Chen C, Zhu Q, Shu M, An ECG Classification Method Based on Multi-Task Learning and CoT Attention Mechanism. Healthcare, 2023, https://doi.org/10.3390/healthcare11071000.

[17] Hassaballah M, Wazery Y, Ibrahim I, Farag A, ECG Heartbeat Classification Using Machine Learning and Metaheuristic Optimization for Smart Healthcare Systems. Bioengineering, 2023, https://doi.org/10.3390/bioengineering10040429.

[18] He K, Zhang X, Ren S, Sun J, Deep Residual Learning for Image Recognition. IEEE Conference on Computer Vision and Pattern Recognition. 90, 770-778, 2015.

[19] Inane O, Giovangrandi L, Kovacs G, Robust Neural-network- based Classification of Premature Ventricular Contractions Using Wavelet Transform and Timing Interval Features. IEEE Trans Biomed Eng, 2006, https://doi.org/10.1109/TBME.2006.880879.
[20] Irfan S, Anjum N, Althobaiti T, Alotaibi A, Siddiqui A, Ramzan N, Heartbeat Classification and Arrhythmia Detection Using a Multi-Model Deep-Learning, 2022 Technique. https://doi.org/10.3390/s22155606.
[21] Kannathal N, Lim C, Rajendra U, Acharya P, Sadasivan K, Cardiac State Diagnosis Using Adaptive Neuro-fuzzy Technique. Medical Engineering & Physics 28, 809–815, 2006.
[21] Kiranyaz S, Ince T, Gabbouj M, Real-time patient-specific ECG classification by 1-D convolutional neural networks. IEEE Transactions on Biomedical Engineering 63(3), 664 – 675, 2016.
[22] Korürek M, Doğan B, ECG beat classification using particle swarm optimization and radial basis function neural network. Expert systems with Applications. 37, 7563-7569, 2010.
[23] Krim H, Tucker D, Donoho H, On Denoising and Best Signal Representation. IEEE Transactions on Information Theory 45, 2225 – 2238, 1999.
[24] Lagerholm M, Peterson C, Braccini G, Ebendrandt L, Sornmo L, Clustering ECG Complexes Using Hermite Functions and Self-organizing Maps. IEEE Trans Biomed Eng 47, 38–48, 2000.
[25] Li H, Yuan D, Wang Y, Cui D, Cao L, Arrhythmia classification based on multi-domain feature extraction for an ECG recognition system, 2016, https://doi.org/10.3390/s16101744.
[26] Li J, Ke L, Du Q, Ding X, Chen X, Research on the Classification of ECG and PCG Signals Based on BiLSTM-GoogLeNet-DS. Applied Sciences, 2022, https://doi.org/10.3390/app122211762.
[27] Mallat S, A Theory of Multiresolution Signal Decomposition: The Wavelet Representation. IEEE Trans: Pattern Anal, Machine Learning 11, 674-693, 1998.
[28] Mathunjwa B, Lin Y, Lin C, Abbod M, Sadrawi M, Shieh J, ECG Recurrence Plot-Based Arrhythmia Classification Using Two-Dimensional Deep Residual CNN Features. Sensors, 2022, https://doi.org/10.3390/s22041660.
[29] Melgani F, Bazi Y, Classification of Electrocardiogram Signals with Support Vector Machines and Particle Swarm Optimization. IEEE Transactions on Information Technology in Biomedicine 12, 667-677, 2008.
[30] Mirza A, Nurmaini S, Partan R, Automatic Classification of 15 Leads ECG Signal of Myocardial Infarction Using One Dimension Convolutional Neural Network. Applied Sciences, 2022, https://doi.org/10.3390/app12115603.
[31] Mishra A, Dharahas G, Gite S, Kotecha K, Koundal D, Zaguia A, Kaur M, Lee H, ECG Data Analysis with Denoising Approach and Customized CNNs. Sensors, 2022, https://doi.org/10.3390/s22051928.
[32] Osowski S, Linh T, ECG Beat Recognition Using Fuzzy Hybrid Neural Network. IEEE Transactions on Biomedical Engineering 48, 1265-1271, 2001.
[33] Peng Z, Chu F, Application of the wavelet transform in machine condition monitoring and fault diagnostics: a review with bibliography. Mechanical systems and signal processing 18, 199-221, 2004.
[34] Pike H, Eilevstjønn H, Bjorland P, Linde J, Ersdal H and Rettedal S, Heart rate detection properties of dry-electrode ECG compared to conventional 3-lead gel-electrode ECG in newborns. BMC Res Notes, 2021, https://doi.org/10.1186/s13104-021-05576-x.
[35] Qu Y, Zhang N, Meng Y, Qin Z, Lu Q, Liu X, ECG Heartbeat Classification Detection Based on WaveNet-LSTM. Sensors Technologies (ICFST), 2020, https://doi.org/10.1109/ICFST51577.2020.9294765.
[36] Rawi A, Elbashir M, Ahmed A, ECG Heartbeat Classification Using CONVXGB Model. Electronics, 2022, https://doi.org/10.3390/electronics11152280.
[37] Russo V, Caturano A, Guerra F, Migliore F et al, Subcutaneous versus transvenous implantable cardioverter defibrillator among drug induced type 1 ECG pattern Brugada syndrome: a propensity score matching analysis from IBRYD study. Heart and Vessels 38:680–688, 2023.
[38] Samesima N, Lazar Neto F, Abrahão Hajjar L et al, Usefulness of ECG criteria to rule out left ventricular hypertrophy in older individuals with true left bundle branch block: an observational study. BMC Cardiovascular Disorders, 2021 https://doi.org/10.1186/s12872-021-02332-8.
[39] Smigiel S, Pałczy´nski K, Ledzi´nski D, Deep Learning Techniques in the Classification of ECG Signals Using R-Peak Detection Based on the PTB-XL Dataset. Sensors, 2021, https://doi.org/10.3390/s21248174.
[40] Wang C, Wei C, Tsai C, Lee Y, Chen K, Lin Y and Lin P, Early detection of myocardial ischemia in resting ECG: analysis by HHT. BioMedical Engineering OnLine, 2023, https://doi.org/10.1186/s12938-023-01089-9.
[41] Wang Y, Wang Y, Zi H, Enhancement of Signal Denoising and Multiple Fault Signatures Detecting in Rotating Machinery Using Dual-Tree Complex Wavelet Transform. Mechanical Systems and Signal Processing 24, 119-137.
[42] Wen T, Lin K, Chang C, Huang H, Classification of ECG Complexes Using Self-organizing CMAC, Measurement 42, 399-407, 2009.
[43] Xiao Q, Lee K, Mokhtar S, Ismail I, Zhang Q, Lim P.Y, Deep Learning-Based ECG Arrhythmia Classification: A Systematic Review, Applied Sciences, 2023, https://doi.org/10.3390/app13084964.
[44] Xiong Z, Stiles M.K, Gillis A.M, Zhao J, Enhancing the detection of atrial fibrillation from wearable sensors with neural style transfer and convolutional recurrent networks, Comput. Biol. Med, 2022, https://doi.org/10.1016/j.compbiomed.2022.105551
[45] Xu1 Y, Hrybouski S, Paterson D, Li D, Lan Y, Luo L, Shen X and Xu L, Comparison of epicardial adipose tissue volume quantification between ECG-gated cardiac and non-ECG-gated chest computed tomography scans. BMC Cardiovascular Disorders, 2022, https://doi.org/10.1186/s12872-022-02958-2.

پنجمین کنفرانس ملی انفورماتیک ایران
پژوهشگاه دانشهای بنیادی، پردیس فرمانیه، تهران
۱۳ و ۱۴ دی ماه ۱۴۰۲

سامانه سلامت سنجی حسگرهای درون خودرویی مبتنی بر شبکه عصبی خودرمزگذار و رگرسیون جنگل تصادفی: نمونه موردی سایپا

سحر ترک حصاری[۱]، بهنام یوسفی مهر[۲]، مهدی قطعی[۳]

[۱] کارشناس ارشد مهندسی فناوری اطلاعات، دانشکده مدیریت، علم و فناوری، دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران)،
s.hesari@aut.ac.ir

[۲] دانشجوی دکتری علوم کامپیوتر، دانشکده ریاضی و علوم کامپیوتر، دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران)،
behnam.y2010@aut.ac.ir

[۳] استاد گروه علوم کامپیوتر، دانشکده ریاضی و علوم کامپیوتر، دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران)،
ghatee@aut.ac.ir

چکیده

با پیشرفت سریع فناوری حسگر و شبکه، پردازش داده‌های حسگرهای سنجش لرزش، دما، فشار، ولتاژ و سایر پارامترهای الکتریکی و مکانیکی خودروها برای کنترل و هشداردهی مورد توجه قرار گرفته است. لازمه ابقای این خدمات، حفظ سلامت حسگرها و تعویض به موقع آنها در هنگام تشخیص خرابی است. در این پژوهش، یک سامانه سلامت سنجی حسگرهای واحد کنترل الکترونیکی خودرو (ECU) پیشنهاد می‌شود که با استفاده از روشهای یادگیری ماشین، مقادیر حسگرها را به صورت مداوم پیشبینی می‌کند. سپس این مقادیر را با مقادیر ثبت شده حسگرها مقایسه نموده و بر اساس یک مدل آماری مبتنی بر توزیع گوسی، خرابی‌های احتمالی را شناسایی می‌کند. با کمک این سامانه، تشخیص زودهنگام خرابی و جایگزینی حسگر خراب، ممکن خواهد بود و این امر موجب کاهش هزینه‌های تعمیر و نگهداری، بهبود تصمیم‌گیری صحیح واحد کنترل خودرو، حفظ نسبت سوخت و هوا در موتور و کاهش آلاینده‌های زیست محیطی می‌شود. کارایی سامانه پیشنهادی، روی داده‌های بیست حسگر مؤثر در واحد کنترل الکترونیکی خودروی کوییک سایپا مورد ارزیابی قرار گرفته است و به کارگیری شبکه عصبی خودرمزگذار برای شناسایی خرابی و رگرسیون جنگل تصادفی برای تخمین مقدار صحیح حسگر و جانشانی آن با مقادیر نادرست حسگر خراب، اثبات شده است. دقت سامانه سلامت سنجی پیشنهادی، روی داده‌های واقعی ذکر شده، ۹۹ درصد بوده است.

کلمات کلیدی

پیش‌بینی خرابی، تشخیص عیب حسگر، خودروی هوشمند، خودرمزگذار، رگرسیون جنگل تصادفی، هوش مصنوعی.

۱- مقدمه

در دنیای امروزی، دستگاه‌هایی با حسگرهای مختلف برای کنترل در همه جای دنیا وجود دارند و تلاش برای ساخت حسگرهای دقیق‌تر و روش‌های جایگزین برای نظارت بر سخت افزارهای مربوطه درحال‌توسعه هستند [1]. علاوه بر این، دائماً روش‌های جدید به منظور شناسایی مشکلات حسگرها به صورت بر خط و حفظ عملکرد سامانه ارائه می‌شوند. موتورهای احتراق داخلی، نمونه بارز دستگاه‌هایی با حسگرهای فراوان و ابزار کنترلی هستند و شناسایی، مکان‌یابی و جداسازی عیوب به یکی از اجزای ضروری طراحی خودرو تبدیل‌شده است[2].

واحد کنترل الکترونیکی موتور[1] در یک موتور احتراق داخلی میزان جریان سوخت تزریقی را برای به دست آوردن مخلوط سوخت و هوای بهینه محاسبه می‌کند و همچنین زمان احتراق مخلوط را با توجه به سایر ویژگی‌های کنترل موتور تعیین می‌کند تا حداکثر گشتاور را در حین احتراق فراهم کند. خوانش متغیرهای سامانه، مانند جریان هوای ورودی به موتور، باز شدن دریچه گاز، سرعت موتور و سایر متغیرها، به واحد کنترل الکترونیک اجازه می‌دهد این محاسبات را انجام دهد. ازآنجایی‌که حسگرها به دلایل مختلف، مستعد خرابی هستند و با ازکارافتادن یک یا چند حسگر، واحد کنترل خودرو به‌طور نادرست، احتراق، تزریق و جریان سوخت را تعیین می‌کند، این امکان وجود دارد که موتور احتراق داخلی به‌طور ناکارآمد کار کرده، ناپایدار یا متوقف شده و در نهایت خراب گردد [3].

این تحقیق جهت تشخیص و جداسازی خطا از یک شبکه عصبی مصنوعی خودرمزگذار [4] برای تخمین مقادیر حسگرها استفاده

[1] Engine control unit (ECU)

می‌کند و هنگامی‌که خرابی رخ می‌دهد، سامانه تشخیص داده و مقدار نزدیک به واقعی آن را با برآوردهای موجود به‌دست‌آمده از سایر حسگرها جایگزین می‌کند.

نوآوری این سامانه این است که برای اولین بار از ایده شبکه خودرمزگذار برای تخمین مقادیر حسگرهای خودرویی استفاده نموده است و آستانه‌ای برای سلامت‌سنجی حسگرهای مختلف خودرو با کمک توزیع گوسی به کار برده است. سامانه پیشنهادی باعث می‌شود بدون هیچ‌گونه افزونگی سخت‌افزاری، خرابی‌های حسگرها شناسایی شده و حتی به کاهش تعداد حسگرها در آینده طراحی‌های مهندسی خودرو منجر شود.

۲- کارهای مرتبط

پیش‌بینی خرابی یک حسگر و جایگزینی یا حل مشکل آن یک فرآیند بسیار پیچیده است، بنابراین عیب‌یابی قبل از بروز مشکل، موضوع پژوهشی است که توجه محیط‌های دانشگاهی و صنعتی را به خود جلب کرده است و هدف آن شناسایی، تشخیص، جداسازی و حل مشکل پیش‌آمده، می‌باشد.

در تشخیص خرابی حسگرهای درون خودروها، روش‌های مختلفی مبتنی بر بررسی افراد خبره، افزودن سخت افزارهای شناساگر یا هوش مصنوعی به کار گرفته می‌شوند. به‌عنوان مثال، در مرجع [5]، یک سامانه تشخیص و جداسازی عیب بر مبنای فیلتر کالمن برای حسگرهای فشار و دمای هوای ورودی منیفولد موتور احتراق داخلی طراحی‌شده است. همچنین، در مرجع [6]، یک روش کنترل خطا با استفاده از کنترل تطبیقی پیشنهاد شده است که برای توربو شارژر، موتور دیزل و گردش مجدد گازهای خروجی استفاده می‌شود. مرجع [7] نیز راهبرد کنترل تحمل خطا را برای دریچه الکترونیکی گاز با استفاده از برآوردگر شبکه‌ی عصبی تطبیقی ارائه کرده است. مرجع [8] یک کنترل تحمل‌پذیر خطای فعال برای کنترل نسبت هوا به سوخت در موتورهای احتراق داخلی ارائه نموده است و از یک ناظر مبتنی بر رگرسیون خطی برای شناسایی، جداسازی، پیکربندی مجدد و کنترل بازخورد متناسب برای حفظ نسبت هوا به سوخت بهره برده است.

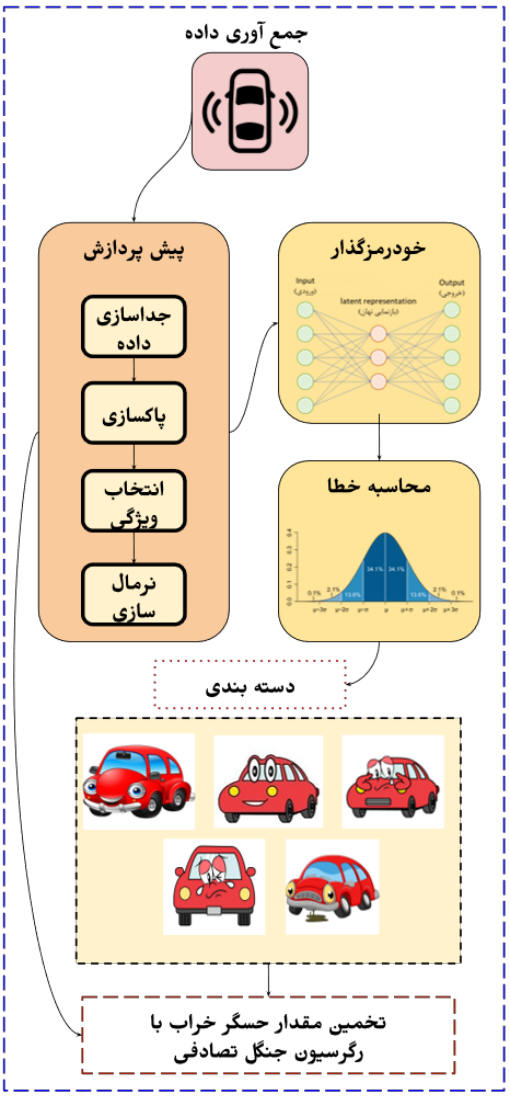
مراجع [9-11] نیز با استفاده از افزونگی سخت‌افزاری و تجزیه‌وتحلیل برای ساخت کنترل تحمل‌پذیر خطای فعال در یک موتور با چندین کنترلر، به جداسازی خطا در حسگرها و محرک‌ها پیشنهاد کرده‌اند. مرجع [12] یک سامانه کنترل فعال مقاوم در مقابل خطا، مبتنی بر شبکه‌های عصبی مصنوعی برای کنترل نسبت سوخت به هوا در موتور جرقه‌زنی ارائه کرده است که پایداری سامانه را حتی در صورت وجود خرابی نشان می‌دهد. مراجع [۱۳-۱۹] نیز از الگوریتم‌ها و روش‌های مختلف یادگیری ماشین و مخصوصاً معماری‌های مختلف شبکه عصبی برای جداسازی خطاها در حسگرها و محرک‌های مختلف استفاده کرده‌اند.

بر اساس جمع‌بندی کارهای پیشین، مشخص شده است که کارهای کمی در رابطه با تشخیص عیب‌های متعدد در موتورهای احتراق داخلی وجود دارد. مراجعی که به بررسی چندین خرابی پرداخته‌اند، استفاده از افزونگی سخت‌افزار در حسگرها را برای مقابله با خرابی‌های متعدد پیشنهاد کرده‌اند که هزینه بالایی دارد. همچنین داده‌هایی که در این تحقیقات استفاده‌شده است، در محیط آزمایشگاهی و در شرایط محیطی ثابت جمع‌آوری‌شده است و محدود به تعداد کمی از حسگرها هستند که این موضوعات باعث ضعف مدل در دنیای واقعی، علاوه بر در نظر گرفته نشدن ارتباط خیلی از حسگرها با یکدیگر خواهند شد.

درحالی‌که در تحقیق حاضر از داده‌های واقعی که در حین رانندگی در شرایط مختلف محیطی جمع‌آوری‌شده است، استفاده می‌شود و همچنین جهت تشخیص خطا از شبکه عصبی مصنوعی خودرمزگذار در حسگرهای موتور احتراق داخلی برای کشف ارتباطات بین ویژگی‌های مختلف بهره گرفته می‌شود. همچنین در اکثر مطالعات قبلی علاوه بر کم بودن تعداد حسگرها، تعدادی از حسگرها به‌عنوان ویژگی و تعدادی به‌عنوان تابع هدف در نظر گرفته شده است. ولی سامانه پیشنهادی در این تحقیق، شامل این محدودیت نبوده و ارتباط همه ویژگی‌ها باهم در نظر گرفته می‌شوند.

۳- سامانه پیشنهادی

سامانه پیشنهادی برای سنجش سلامت حسگرهای خودرو در شکل ۱ ارائه شده است. این سامانه مبتنی بر اصول داده‌کاوی و یادگیری ماشین بنا شده و امکان بررسی موازی وضعیت حسگرهای خودرو را فراهم می‌نماید.



شکل (۱): معماری سامانه پیشنهادی

در این سامانه ابتدا داده‌ها جمع‌آوری‌شده، بعد از انجام پیش‌پردازش‌های لازم روی داده‌های خام، با استفاده از شبکه عصبی خودرمزگذار مقادیر حسگرها تخمین زده‌شده و با به دست آوردن اختلاف مقدار تخمین زده‌شده مدل و مقدار واقعی حسگر و مقایسه اختلاف به‌دست‌آمده با حد آستانه تعریف‌شده برای هر حسگر، میزان سلامت حسگرها به‌صورت کیفی به دسته‌های سالم، سالم باکمی خطا، تقریباً سالم، تقریباً خراب و خراب تفکیک می‌نماید. درنهایت با استفاده از رگرسیون جنگل تصادفی مقدار حسگر خراب تخمین زده و جایگزینی می‌شود که در ادامه نحوه‌ی کار و اجزا تشکیل‌دهنده هر بخش شرح داده می‌شود.

۳-۱- جمع‌آوری داده

شرکت سایپا یدک با همراهی اپراتور ایرانسل و شرکت دلفین آپادانا پروژه خودروهای متصل را راه‌اندازی کرده است. شرکت دلفین آپادانا بردهایی را ساخته است که از طریق سوکت عیب‌یاب به واحد کنترل الکترونیکی متصل می‌شود و داده‌های حسگرهای مختلف را از آن دریافت می‌کند و در بازه‌های زمانی یک دقیقه تا ده دقیقه از طریق سیم‌کارتی که روی آن تعبیه‌شده است بر روی سرورهای اختصاصی شرکت سایپا یدک ارسال و ذخیره می‌کند و از طریق نرم‌افزار همراه که متعلق به شرکت دلفین آپادانا می‌باشد و روی گوشی رانندگان نصب‌شده، موقعیت جغرافیایی خودرو و همچنین سرعت و خطاهای ثبت‌شده در واحد کنترل الکترونیکی خودرو را در آن نمایش می‌دهد؛ این در حالی است که قبلاً خطاهای واحد کنترل الکترونیکی فقط از طریق دستگاه عیب‌یابی قابل‌دسترس بود.

در این پژوهش از موتور احتراق داخلی خودروی کوییک استفاده‌شده و با استفاده از برنامه مذکور، داده‌های حسگرهای مختلف از واحد کنترل الکترونیکی خودرو دریافت، ثبت و استفاده‌شده است.

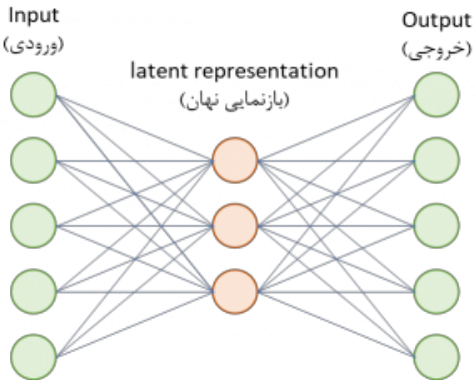
۳-۲- پیش‌پردازش

در این بخش ابتدا به جداسازی داده می‌پردازیم، به این نحو که در کل ۳۳ درصد داده‌ها به‌عنوان داده ارزیابی و ۶۷ درصد داده‌ها به‌عنوان داده آموزش در نظر گرفته‌شده است و جهت به دست آوردن آستانه خطا، ۲۵ درصد داده‌های آموزش به داده اعتبارسنجی اختصاص داده‌شده است.

پس از جداسازی داده‌ها به پاک‌سازی و نرمال‌سازی داده‌ها می‌پردازیم. پاک‌سازی داده‌ها، فرآیند شناسایی، حذف یا تصحیح سوابق اشکال‌دار یا نادرست از یک مجموعه دادگان است و به شناسایی قسمت‌های ناقص، نادرست یا نامربوط از داده‌ها و سپس جایگزینی، اصلاح یا پاک کردن این نوع داده‌ها می‌پردازد. درروش پیشنهادی، ابتدا مجموعه دادگان ازلحاظ داده‌های نویزی و تکراری بررسی‌شده و در صورت وجود، این نوع داده‌ها حذف می‌شوند. سپس به انتخاب ویژگی‌های مربوط به حسگرها از کل مجموعه جمع‌آوری‌شده پرداخته و درنهایت به نرمال‌سازی داده‌ها پرداخته می‌شود. نرمال‌سازی داده‌ها، روشی است که برای استانداردسازی دامنه ویژگی‌های داده استفاده می‌شود. ازآنجایی‌که دامنه مقادیر داده‌ها ممکن است بسیار متفاوت باشد، این یک مرحله ضروری در پیش‌پردازش داده‌ها در حین استفاده از الگوریتم‌های یادگیری ماشین است، زیرا الگوریتم‌های یادگیری ماشین فقط اعداد را می‌بینند و اگر تفاوت زیادی در دامنه وجود داشته باشد، ممکن است این فرض اساسی را ایجاد کند که اعداد با محدوده بالاتر به‌نوعی برتری دارند. در این پژوهش از نرمال‌سازی مین مکس استفاده می‌شود.

۳-۳-خودرمزگذار

جهت تخمین خطا از یک شبکه عصبی خودرمزگذار[4] بهره می‌بریم، معماری این مدل در شکل ۲ قابل مشاهده است.



شکل (۲): خودرمزگذار[4]

برای تخمین خطا، داده‌های حسگرها را به‌عنوان ورودی و خروجی مدل در نظر می‌گیریم و انتظار داریم مدل از طریق مدل خودرمزگذار و لایه میانی به وزن دهی ارتباط حسگرها بپردازد و بتواند از روی لایه میانی ساخته‌شده به ساخت لایه خروجی بپردازد. در نتیجه با ارسال داده‌های حسگر به مدل و ساخت لایه خروجی توسط مدل و مقایسه مقدار واقعی حسگر با مقدار پیش‌بینی‌شده توسط مدل، به میزان سلامت حسگر پی برده شود.

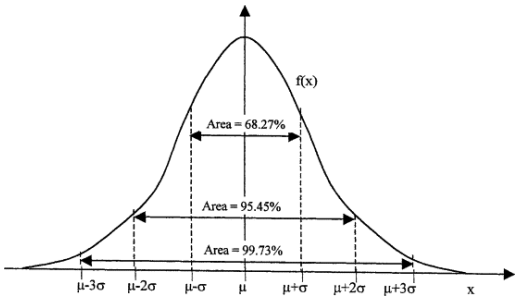
به همین دلیل با استفاده از یک شبکه عصبی خودرمزگذار، بیست عدد از حسگرهایی که بیشترین ارتباط را با یکدیگر داشتند، به‌عنوان ورودی خودرمزگذار انتخاب شدند و با استفاده از یک‌لایه میانی با دوازده گره عصبی، داده‌ها به بعد پایین‌تر نگاشته شده و با استفاده از لایه میانی به تولید خروجی پرداخته شد. پس از تولید داده در لایه خروجی انتظار می‌رود که داده به‌دست‌آمده به داده اولیه تا حد امکان نزدیک باشد و با فاصله گرفتن این دو متغیر درمی‌یابیم که حسگر مربوطه دچار خطا شده است.

۳-٤-محاسبه مقدار خطا

هنگام دریافت داده از حسگرها (داده‌های واقعی) و عبور داده‌های حسگرها از مدل خود رمزگذار، مقدار تخمین زده‌شده مدل به دست می‌آید که با محاسبه قدر مطلق تفریق مقدار واقعی و مقدار تخمین زده، اختلاف این دو متغیر به دست می‌آید. سپس با فرض اینکه داده‌های خطا از توزیع نرمال تبعیت کنند،

انحراف معیار را روی داده اعتبارسنجی محاسبه کرده و به‌عنوان آستانه‌ای برای دسته‌بندی خطا استفاده می‌کنیم.

انحراف معیار یکی از شاخص‌های پراکندگی است که نشان می‌دهد به‌طور میانگین داده‌ها چه مقدار از مقدار متوسط فاصله دارند. همان‌طور که در شکل ۳ مشاهده می‌شود، افزایش انحراف معیار به این معنی است که درصد کمتری از داده در این ناحیه از توزیع مشاهده‌شده است و احتمال خطا بالاتر می‌رود.



شکل (۳): درصدهای مختلف داده در توزیع نرمال[21]

در نتیجه پس از محاسبه انحراف معیار و فاصله داده از حول میانگین، میزان سلامت حسگرها مطابق جدول ۱ به دسته‌های سالم، سالم باکمی خطا، تقریباً سالم، تقریباً خراب و خراب تفکیک می‌شوند.

جدول (۱) : شاخص‌های تعیین سلامت

اختلاف مقدار (E)	شاخص تعیین سلامت
$\mu-\sigma < E < \mu+\sigma$	سالم
$\mu\pm\sigma \leq E \leq \mu\pm2\sigma$	سالم با کمی خطا
$\mu\pm2\sigma \leq E \leq \mu\pm3\sigma$	تقریباً سالم
$\mu\pm3\sigma \leq E \leq \mu\pm4\sigma$	تقریباً خراب
$E > \mu\pm4\sigma$	خراب

۳-۵-تخمین مقدار حسگر خراب

پس از اینکه تشخیص داده شد که یک حسگر دچار خرابی شده است، علاوه بر اینکه مقدار تخمین زده‌شده با استفاده از شبکه خودرمزگذار قابل‌جایگزینی با مقدار حسگر خراب است اما به دلیل عدم دقیق بودن مدل خودرمزگذار برای تخمین در برخی از حسگرها، از یک رگرسیون جنگل تصادفی [22] بهره برده‌ایم و با ترکیب این دو مدل و به‌کارگیری سایر حسگرها به‌عنوان ویژگی و در نظر گرفتن همبستگی میان حسگرهای مختلف، مقداری نزدیک‌تر به مقدار واقعی تخمین زده‌شده و جایگزین مقدار حسگر خراب می‌شود. شایان ذکر است که آموزش مدل جنگل تصادفی مجزا بوده و فقط روی داده آموزش پس از فرآیند پیش‌پردازش رخ می‌دهد.

۴- نتایج پیاده‌سازی

برای ارزیابی سامانه پیشنهادی از ضریب تعیین[2] استفاده‌شده است. ضریب تعیین یک معیار آماری است که در تحلیل رگرسیون استفاده می‌شود تا نشان‌دهنده‌ی میزان تطابق مدل رگرسیون با داده‌های واقعی باشد. این ضریب معمولاً بین صفر و یک قرار می‌گیرد و مقدار بالاتر نشان‌دهنده تطابق بهتر مدل با داده‌ها است.

فرمول ضریب تعیین به‌صورت زیر است:

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2} \qquad (۱)$$

که پارامترهای آن به شرح زیر هستند:

- ضریب تعیین: R^2
- مقادیر واقعی مشاهده‌شده در داده‌ها: y_i
- مقادیر پیش‌بینی‌شده توسط مدل رگرسیون: $\hat{y}_i$
- میانگین مقادیر y در داده‌ها: $\bar{y}$

بر اساس نتایج به دست آمده روی داده های کوییک، مقدار ضریب تعیین به‌دست‌آمده روی کل ۲۰ حسگر موجود در مجموع دادگان بالای ۹۹ درصد بوده است. جدول ۲ سامانه پیشنهادی را با کارهای اخیر مرتبط مقایسه می‌کند.

جدول (۲): مقایسه سامانه پیشنهادی با کارهای اخیر

مرجع	سال انتشار	تعداد حسگر	تعداد مدل	ضریب تعیین
[8]	۲۰۱۹	۴	۴	۰.۸
[15]	۲۰۲۰	۲	۳	۰.۹۳
[19]	۲۰۲۳	۳	۵	۰.۹۹
سامانه پیشنهادی	۲۰۲۳	۲۰	۲	۰.۹۹

همان‌طور که مشاهده می‌شود سامانه پیشنهادی توانسته روی داده‌های غیر آزمایشگاهی و تعداد خیلی زیادی حسگر با پیچیدگی محاسباتی کمتر به دقت فوق‌العاده‌ای برسد.

۵- نتیجه‌گیری و کارهای آتی

در این مقاله، یک سامانه پیشنهادی برای پیش‌بینی خرابی و تخمین مقدار حسگرهای درون خودرویی با تکیه بر شبکه عصبی خودرمزگذار و رگرسیون جنگل تصادفی مطرح شد. معماری سامانه به‌گونه‌ای است که داده‌های ارسالی از حسگرها را به‌عنوان ورودی پذیرفته و با استفاده از آستانه خطا تعریف‌شده و نتیجه مدل خودرمزگذار، برای هر حسگر شاخص سلامت تعیین می‌کند و در صورت خرابی حسگر با استفاده از مدل رگرسیون جنگل تصادفی، تخمینی برای حسگر معیوب جایگزین می‌کند. شایان ذکر است که سامانه پیشنهادی روی‌داده‌هایی که از خودروها در حین رانندگی و در محیط واقعی توسط شرکت سایپا یدک جمع‌آوری‌شده است، آموزش و آزمایش‌شده است و مدلی واقعی و بدون سوگیری ارائه‌شده است.

[2] Coefficient Of Determination

به‌عنوان کار آتی پیشنهاد می‌شود روی آستانه خرابی پیشنهادی برای شاخص‌های تعیین سلامت بررسی جامع‌ترصورت گرفته و دقت آستانه خرابی اثبات گردد.

سپاسگزاری

نویسندگان مقاله، مراتب تشکر و قدردانی خود به شرکت سایپا یدک و شرکت دلفین آپادانا به دلیل فراهم کردن مجموع دادگان مورد استفاده این تحقیق و تلاش برای ارتقای فناوری‌های خودرویی تقدیم می‌نمایند.

مراجع

[1] Dereszewski, Mirosław, and Grzegorz Sikora. "Diagnostics of the internal combustion engines operation by measurement of crankshaft instantaneous angular speed." Journal of KONBiN 49.4 (2019): 281-295.
[2] Mehranbod, Nasir, Masoud Soroush, and Chanin Panjapornpon. "A method of sensor fault detection and identification." Journal of Process Control 15.3 (2005): 321-339.
[3] Suwatthikul, Jittiwut. "Fault detection and diagnosis for in-vehicle networks." Fault Detection (2010): 283-306.
[4] Hinton, Geoffrey E., and Ruslan R. Salakhutdinov. "Reducing the dimensionality of data with neural networks." science 313.5786 (2006): 504-507.
[5] Carbot-Rojas, Diego A., Gildas BesanÇon, and Ricardo Fabricio Escobar-Jiménez. "EKF based sensor fault diagnosis for an internal combustion engine." 2019 23rd International Conference on System Theory, Control and Computing (ICSTCC). IEEE, 2019.
[6] Murtaza, Ghulam, Aamir I. Bhatti, and Yasir A. Butt. "Super twisting controller-based unified FDI and FTC scheme for air path of diesel engine using the certainty equivalence principle." Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering 232.12 (2018): 1623-1633.
[7] Li, Shoutao, et al. "Friction fault diagnosis and fault tolerant control for electronic throttles with sliding mode and adaptive RBF estimator." Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering 235.9 (2021): 1605-1614.
[8] Amin, Arslan Ahmed, and Khalid Mahmood-ul-Hasan. "Robust active fault-tolerant control for internal combustion gas engine for air–fuel ratio control with statistical regression-based observer model." Measurement and Control 52.9-10 (2019): 1179-1194.
[9] Amin, Arslan Ahmed, and Khalid Mahmood-ul-Hasan. "Unified fault-tolerant control for air-fuel ratio control of internal combustion engines with advanced analytical and hardware redundancies." Journal of Electrical Engineering & Technology 17.3 (2022): 1947-1959.
[10] Amin, Arslan Ahmed, and Khalid Mahmood-ul-Hasan. "Robust passive fault tolerant control for air fuel ratio control of internal combustion gasoline engine for sensor and actuator faults." IETE Journal of Research 69.5 (2023): 2846-2861.
[11] Amin, Arslan Ahmed, and Khalid Mahmood-Ul-Hasan. "Advanced fault tolerant air-fuel ratio control of internal combustion gas engine for sensor and actuator faults." IEEE Access 7 (2019): 17634-17643.
[12] Shahbaz, Muhammad Hamza, and Arslan Ahmed Amin. "Design of active fault tolerant control system for air fuel ratio control of internal combustion engines using artificial neural networks." IEEE Access 9 (2021): 46022-46032.
[13] Guzmán-Zaragoza, M., et al. "Fault detection and isolation in sensors of an internal combustion engine." Memorias del Congreso Nacional de Control Automático. 2020.
[14] Mofleh, Ahmed F., Ahmed N. Shmroukh, and Nouby M. Ghazaly. "Fault detection and classification of spark ignition engine based on acoustic signals and artificial neural network." International Journal of Mechanical and Production Engineering Research and Development 10.3 (2020): 5571-5578.
[15] 13. Yu, D.L., et al., Dynamic fault detection and isolation for automotive engine air path by independent neural network model. 2014. 15(1): p. 87-100.
[16] Sangha, M. S., et al. "Fault detection and identification of automotive engines using neural networks." IFAC Proceedings Volumes 38.1 (2005): 272-277.
[17] Ghazaly, Nouby M., et al. "Prediction of misfire location for SI engine by unsupervised vibration algorithm." Applied Acoustics 192 (2022): 108726.
[18] Wang, Y. S., et al. "An engine-fault-diagnosis system based on sound intensity analysis and wavelet packet pre-processing neural network." Engineering applications of artificial intelligence 94 (2020): 103765.
[19] Cervantes-Bobadilla, M., et al. "Multiple fault detection and isolation using artificial neural networks in sensors of an internal combustion engine." Engineering Applications of Artificial Intelligence 117 (2023): 105524.
[20] Priya Varshini, A. G., and K. Anitha Kumari. "Predictive analytics approaches for software effort estimation: A review." Indian J. Sci. Technol 13 (2020): 2094-2103.
[21] Jain, S.K., and V.P. Singh. "Chapter 4 - Statistical Techniques for Data Analysis." Developments in Water Science, edited by S.K. Jain and V.P. Singh, vol. 51, Elsevier, 2003, pp. 207-276. ISSN 0167-5648.
[22] Breiman, Leo. "Random forests." Machine learning 45 (2001): 5-32.

پنجمین کنفرانس ملی انفورماتیک ایران
پژوهشگاه دانشهای بنیادی، پردیس فرمانیه، تهران
۱۳ و ۱۴ دی ماه ۱۴۰۲

مدیریت منابع مبتنی بر نظریه بازی برای کاربردهای بی‌درنگ با استفاده از Lévy Walk در سامانه‌های لبه

ابوالفضل یونسی[۱] و محسن انصاری[۲]

[۱] دانشجوی کارشناسی ارشد، دانشکده‌ی مهندسی کامپیوتر، دانشگاه صنعتی شریف
تهران، ایران
abolfazl.yunesi@sharif.edu

[۲] استادیار، دانشکده‌ی مهندسی کامپیوتر، دانشگاه صنعتی شریف
تهران، ایران
ansari@sharif.edu

چکیده

در سال‌های اخیر، محاسبات لبه موبایل (Mobile Edge Computing) به‌عنوان یک راه‌حل مناسب برای پشتیبانی از برنامه‌های کاربردی مهم در راستای بهبود تأخیر و بهبود کیفیت خدمات در شبکه‌های نسل بعدی ظهور کرده است. بااین‌حال، تغییرات هم‌بندی لبه پویا ناشی از تحرک گره، چالش‌های مدیریت منابع قابل‌توجهی را ایجاد می‌کند. رویکردهای موجود معمولاً به کنترل متمرکز یا زیرساخت‌های استاتیک متکی هستند. این مقاله یک الگوریتم نظریه‌ی بازی جدید را برای تخصیص منابع توزیع‌شده در سامانه‌های لبه تلفن همراه پیشنهاد می‌کند که از برنامه‌های اینترنت اشیاء بی‌درنگ پشتیبانی می‌کنند. الگوریتم پیشنهادی که از Lévy walk تقلید می‌کند، حرکات گره لبه را مدل می‌کند. در هر شکاف زمانی، گره‌ها با همسایگان از طریق چانه‌زنی محلی بر اساس پیشنهادها و پاسخ‌های موقت، با همسایگان مذاکره می‌کنند. این هماهنگی توزیع‌شده واقع‌بینانه بدون ساماندهی متمرکز را تقلید می‌کند. درواقع، در این مقاله از الگوریتم پیشنهادی تخصیص منابع توزیع‌شده مبتنی بر Lévy walk (DR2A) برای حل مسئله بهینه‌سازی مشترک و همگرایی مکرر به سبب تخصیص‌های تعادل نش استفاده می‌شود. شبیه‌سازی‌های انجام‌شده، الگوریتم را در شدت‌های بارکاری و چگالی گره‌های مختلف ارزیابی می‌کنند. نتایج نشان می‌دهد نسبت پذیرش وظیفه به‌طور قابل‌توجهی بیشتر از ۴% در مقایسه با مدل‌های لبه ایستا و ابر، با کاهش تأخیر حداقل ۶% و صرفه‌جویی در انرژی حداقل ۲۲% ایجاد کرده است.

کلمات کلیدی

نظریه بازی، مدیریت منابع، مدیریت انرژی، تحرک، تأخیر، قابلیت اطمینان

۱- مقدمه

رشد سریع دستگاه‌ها و فناوری‌های اینترنت اشیا[1] نسل جدیدی از برنامه‌های حساس به تأخیر[2] مانند واقعیت افزوده، وسایل نقلیه خودمختار، مراقبت‌های بهداشتی هوشمند و اتوماسیون صنعتی را قادر ساخته است [۱]. بااین‌حال، پشتیبانی مؤثر از این برنامه‌های کاربردی بی‌درنگ، به دلیل ویژگی‌های منحصربه‌فرد زیرساخت‌های محاسباتی لبه تلفن همراه، چالش‌های عمده‌ای در مدیریت منابع ایجاد می‌کند. گره‌های لبه به دلیل تحرک[3] کاربر، محدودیت‌های شدید منابع و قابلیت دسترس‌پذیری کمتری دارند [۲]. درنتیجه، طراحی الگوریتم‌های تخصیص منابع کارآمد حیاتی است. این الگوریتم‌ها باید قابلیت‌های محاسباتی و ذخیره‌سازی محدود گره‌های لبه را در نظر بگیرند و درعین‌حال از اتصالات با تأخیر کم و قابلیت اطمینان[4] بالا اطمینان حاصل کنند.

علاوه بر این، آن‌ها همچنین باید ماهیت پویای زیرساخت‌های محاسباتی لبه را در نظر بگیرند و با تغییرات در تحرک کاربر و در دسترس بودن گره‌های لبه سازگار شوند [۳].

روش‌های تخصیص منابع متمرکز سنتی به‌خوبی به محیط لبه توزیع‌شده در مقیاس بزرگ نمایان نمی‌شوند و نمی‌توانند با تغییرات هم‌بندی پویا سازگار شوند [۱][۲]. درحالی‌که رویکردهای توزیع‌شده مقیاس‌پذیری و سازگاری را بهبود می‌بخشند، امکان همکاری کارآمد بین گره‌های لبه‌ای که به نفع شخصی هستند، بی‌اهمیت است [۴]. بدون انگیزه‌های مناسب، اشتراک توزیع‌شده منابع خطراتی را به همراه دارد که منجر به تخصیص حریصانه و ناعادلانه می‌شود که در بهینه‌سازی شکست می‌خورد. برای مقابله با این چالش‌ها، محققان ساختارهای انگیزشی مختلفی را در محیط لبه توزیع‌شده پیشنهاد کرده‌اند که در بخش ۲ مروری بر آن‌ها خواهیم داشت. یکی از این مکانیسم‌ها استفاده از نظریه

[1] Internet of Things
[2] Latency-Sensetive
[3] Mobility
[4] Reliability

بازی است که در آن گره‌های لبه به‌عنوان بازیکنان منطقی باهدف به حداکثر رساندن مطلوبیت خود در نظر گرفته می‌شوند [۵]. با طراحی توابع سودمند مناسب و مکانیسم‌های پاداش، می‌توان همکاری را تشویق کرد و از رفتار خودخواهانه در میان گره‌های لبه جلوگیری کرد که منجر به تخصیص منابع منصفانه‌تر و بهبود کارایی سیستم می‌شود. این ساختارهای انگیزشی راه را برای مدیریت مؤثر منابع در محیط لبه توزیع‌شده هموار می‌کنند و ضمن در نظر گرفتن منافع شخصی گره‌ها، استفاده بهینه از منابع را تضمین می‌کنند [۵].

تئوری بازی چارچوبی ریاضی برای مدل‌سازی تعاملات راهبردی[1] بین تصمیم‌گیرندگان منطقی و هدایت آن‌ها به‌سوی نتایج اجتماعی بهینه از طریق رفتارهای خودخواهانه ارائه می‌کند. راه‌حل‌های نظریه بازی موجود برای مدیریت منابع لبه به‌طور کامل از الگوهای تحرک گره‌های لبه استفاده نمی‌کنند که در دسترس بودن منابع در طول زمان به روش‌های غیرقابل‌پیش‌بینی تأثیر می‌گذارد. نادیده گرفتن ویژگی‌های تحرک گره می‌تواند منجر به تصمیمات تخصیص نامناسب و ناپایدار شود [۴][۵][۶]. با ترکیب ویژگی‌های تحرک گره در مدل‌های نظریه بازی، می‌توان به نمایش دقیق‌تری از در دسترس بودن منابع دست‌یافت. این امر به تصمیم‌گیرندگان اجازه می‌دهد تا تصمیمات تخصیص آگاهانه‌تری را در حالت بی‌درنگ اتخاذ کنند که منجر به ثبات بهتر و نتایج بهینه می‌شود. علاوه بر این، درک تأثیر تحرک گره بر تخصیص منابع همچنین می‌تواند توسعه راهبردهای پویا را که با شرایط متغیر سازگار می‌شوند، امکان‌پذیر سازد و کارایی و اثربخشی سامانه‌های مدیریت منابع لبه را بهبود بخشد.

در این مقاله، ما یک چارچوب مدیریت منابع توزیع‌شده مبتنی بر نظریه بازی جدید برای برنامه‌های کاربردی حساس به تأخیر در سامانه‌های محاسباتی لبه چند دسترسی تلفن همراه پیشنهاد می‌کنیم. ما چندین مشارکت کلیدی انجام می‌دهیم که در ادامه لیست شده‌اند.

- ما یک بازی غیر همکاری را برای مدل‌سازی چانه‌زنی توزیع‌شده بین گره‌های لبه موبایل ایجاد می‌کنیم.
- ما از الگوهای Lévy walk برای توصیف واقع‌بینانه تحرک گره و تأثیر آن بر پویایی استفاده می‌کنیم.
- ما یک الگوریتم سبک‌وزن مبتنی بر چانه‌زنی طراحی می‌کنیم که در آن گره‌ها به‌طور مستقل منابع را در مسیرهای Lévy walk بررسی و تخصیص می‌دهند.

۲- کارهای پیشین

نویسندگان در [۸] یک الگوریتم تخلیه وظیفه را برای به حداقل رساندن هزینه انرژی با بهینه‌سازی مشترک نرخ تخلیه وظیفه، توان انتقال و دستگاه‌های تلفن همراه برای بارگیری محاسباتی چند کاربره پیشنهاد می‌کنند. نویسندگان در [۹] روشی را برای بارگذاری امن برنامه‌ها به‌صورت بی‌درنگ پیشنهاد کرد که در دسترس بودن و نزدیکی منابع را در نظر می‌گیرند. آن‌ها همچنین هزینه‌های اجرا و تأخیر آپلود برنامه‌ها در سرورهای لبه را در نظر می‌گیرند. تحقیقات آن‌ها پتانسیل استفاده از سامانه‌های محاسباتی مه برای تخلیه و گسترش ذخیره‌سازی را بررسی می‌کند. بااین‌حال، راه‌حل آن‌ها یک اشکال دارد: نمی‌تواند در دسترس بودن منابع را همیشه تضمین کند، به‌خصوص در مواردی که سرور مه بارگذاری بیش‌ازحد است. در نتیجه، عملکرد سیستم ممکن است به‌تدریج کاهش یابد. پانجاناتان و همکاران [۱۰] یک الگوریتم زمان‌بندی پیشنهاد کردند که از صف کم تأخیر (LLQ) برای بهبود کیفیت خدمات[2] (QoS) برای برنامه‌های صوتی و تصویری، با کاهش تأخیر و افزایش توان استفاده می‌کند. علاوه بر این، ژانگ و همکاران، [۱۱] یک الگوریتم زمان‌بندی منابع برای رسیدگی به درخواست‌های بارگیری چندگانه موبایل به‌طور هم‌زمان پیشنهاد کرد. این الگوریتم با به حداکثر رساندن توان عملیاتی و به حداقل رساندن زمان پردازش برنامه‌های تلفن همراه، منابع را به‌طور مؤثر به چندین کاربر تلفن همراه اختصاص می‌دهد.

ما در این مقاله چندین مزیت نسبت به کارهای پیشین داریم: اولاً، مدل‌سازی ریاضی پیچیده‌تری از Lévy walk ارائه می‌دهیم که می‌تواند برای مدل کردن جریان ترافیک و ارتباطات در شبکه‌های بزرگ کاربردی باشد. دوما، مؤلفه‌های تصادفی و غیرقابل‌پیش‌بینی مانند تأخیر، میزان تقاضا و منابع موردنیاز را شامل می‌شود درحالی‌که بسیاری از مراجع ذکرشده فقط به بهینه‌سازی هزینه‌ها یا عملکرد می‌پردازند. سوما، اثرات متقابل بین اجزای مختلف سیستم را به‌طور ریاضی مدل می‌کند تا رفتار سیستم را در شرایط واقعی‌تری مدل کند.

۳- سیستم مدل

ما یک سامانه رایانش لبه موبایل متشکل از دستگاه‌های اینترنت اشیا، گره‌های لبه و یک سرور لبه را در نظر می‌گیریم.

دستگاه‌های اینترنت اشیا: مجموعه‌ای از دستگاه‌های اینترنت اشیا، وظایف کاربردی حساس به تأخیر را تولید می‌کنند. هر دستگاه اینترنت اشیا یک مجموعه $n \in \mathrm{N}$ دارای الزامات محاسباتی و شبکه‌ای است که توسط یک تاپل (c_n, b_n) تعریف‌شده است که در آن c_n چرخه‌های CPU و b_n حجم داده موردنیاز برای پردازش یک وظیفه است. هر وظیفه k نیاز به پردازش دارد:

$C_{n,k}$ – چرخه‌های CPU موردنیاز، برگرفته از توزیع نرمال:

$$C_{n,k} \sim N(\mu_c, \sigma_c)$$

$B_{n,k}$ – اندازه داده، برگرفته از توزیع log-normal:

$$B_{n,k} \sim LogN(\mu_b, \sigma_b)$$

این تنوع درشدت منابع در انواع مختلف برنامه را نشان می‌دهد. تعداد وظایف $T_{n,t}$ تولیدشده توسط دستگاه n در هر شکاف زمانی t به‌عنوان یک متغیر تصادفی پواسون مدل‌سازی می‌شود:

$$T_{n,t} \sim Poisson(\varphi_n)$$

φ_n در معادله بالا میانگین نرخ تولید برای دستگاه n است. این منعکس‌کننده انبوه کارهایی است که به‌طور مستقل با احتمال مشخص φ_n در هر واحد زمانی وارد می‌شوند. مقادیر φ_n بر اساس پروفایل طولانی‌مدت بارهای کاری از هر نوع دستگاه (به‌عنوان‌مثال حس‌گر دما و غیره) تخمین زده می‌شود. با توجه به φ_n، تعداد خاصی از وظایف $T_{n,t}$ (نشان‌دهنده مدل تولید وظایف برای دستگاه n است که مشخص می‌کند در هر بازه زمانی چه تعداد وظیفه‌ای تولید و به گره‌ها ارسال خواهد شد) تولیدشده در هر شکاف t نمونه‌برداری می‌شود. ورود این وظایف حجم کاری را تشکیل می‌دهد که باید تحت مهلت‌های تأخیر پردازش شود تا سودمندی تعیین شود. این بهینه‌سازی منابع پویا توسط الگوریتم نظری بازی ما انجام می‌شود.

گره‌های لبه: مجموعه‌ای از گره‌های لبه متحرک در سراسر منطقه تحت پوشش شبکه مستقرشده‌اند تا خدمات رایانش ابری را ارائه دهند. برخی گره‌های

[1] Strategic

[2] Quality of Service

لبه مانند m دارای منابع پردازشی (مانند CPU) و ارتباطی (مانند پهنای باند) محدود هستند که باید بین وظایف مختلف تقسیم شوند. میزان منابع محاسباتی و ارتباطی هر گره m محدود به f_m و b_m است. گره‌ها می‌توانند وظایف را به‌صورت محلی پردازش کنند یا به دیگر گره‌ها/ابر بارگذاری کنند.

سرور لبه: یک سرور متمرکز سیستم کلی را با وضعیت منابع جهانی هماهنگ می‌کند اما منابع را مستقیماً تخصیص نمی‌دهد.

۳-۱- مدل‌سازی تحرک با استفاده از Lévy walk

ما رفتار تحرک گره‌های لبه متحرک را با استفاده از Lévy walk در زمان گسسته مدل می‌کنیم. یک فضای دوبعدی را در نظر بگیرید که گره‌ها در آن حرکت می‌کنند. اجازه دهید $(x_{m,t},y_{m,t})$ مختصات گره m را در زمان t نشان دهیم. تکامل مکان‌های گره در شکاف‌های زمانی متوالی $t = ۱, ۲, \dots$ توسط فرآیند تحرک Lévy walk توصیف می‌شود. طول گام‌های $l_{m,t}$ که توسط گره m بین شکاف‌های زمانی متوالی گرفته می‌شود، متغیرهای تصادفی هستند که از توزیع Lévy $L(\lambda)$ گرفته‌شده‌اند. توزیع Lévy یک توزیع احتمال سنگین است که به‌صورت رابطه ۱ تعریف می‌شود:

$$L(\lambda) = \frac{1}{(\lambda t^3)} \; for\; t \geq 0 \qquad (۱)$$

در اینجا $\lambda > 0$ یک پارامتر مقیاس بندی است که تغییر طول گام را تعیین می‌کند. توزیع Lévy طول پله‌های توزیع‌شده تصادفی را با این ویژگی کلیدی تولید می‌کند که احتمال طول پله‌های بزرگ به‌تدریج کاهش می‌یابد (به‌عنوان یک قانون توان) نسبت به توزیع عادی. این منجر به پرش‌های متناوب طولانی‌تر با احتمال غیر صفر می‌شود. روش حرکت گره‌های لبه را به‌صورت تصادفی با الهام از مدل Levy walk طراحی کردیم تا هم ویژگی‌های تصادفی لازم برای مدل‌سازی حرکت غیرقابل‌پیش‌بینی گره‌ها را فراهم کند و هم ویژگی حافظه بلند مشاهده‌شده در داده‌های واقعی تحرک را حفظ نماید؛ بنابراین پارامتر λ را برابر ۱.۵ تنظیم کردیم. طول گام $l_{m,t}$ برای هر گره m به‌طور مستقل از توزیع levy، $L(۱.۵)$ نمونه‌برداری می‌شود. جهت حرکت $\theta_{m,t}$ در هر زمان t به‌عنوان زاویه‌ای که به‌طور احتمالی از توزیع یکنواخت انتخاب‌شده است نشان داده می‌شود:

$$\theta_{m,t} \sim U(0, 2\pi)$$

این امر تصادفی بودن جهت گرفته‌شده در هر مرحله را با حفظ تغییرات طول دم سنگین معرفی می‌کند. با توجه به طول گام $l_{m,t}$ و جهت $\theta_{m,t}$، می‌توانیم مکان جدید $(x_{m,t+1},y_{m,t+1})$ گره m را بعد از زمان t با استفاده از رابطه‌های ۲ و ۳ محاسبه کنیم:

$$x_{m,t+1} = x_{m,t} + l_{m,t} \times cos(\theta_{m,t}) \qquad (۲)$$
$$y_{m,t+1} = y_{m,t} + l_{m,t} \times sin(\theta_{m,t}) \qquad (۳)$$

این عبارات به‌صورت بازگشتی مسیر تصادفی هر گره m را در طول زمان بر اساس فرآیند Lévy walk ترسیم می‌کنند. مدل Lévy walk یک مبنای واقعی و ریاضی دقیق برای شبیه‌سازی تحرک سیال گره‌های لبه در الگوریتم تخصیص منابع و فرمول‌بندی بازی ما فراهم می‌کند. این امکان ارزیابی عملکرد در شرایط طبیعی موبایل را فراهم می‌کند.

۳-۲- مدل بازی غیر مشارکتی

تخصیص منابع توزیع‌شده بین گره‌های لبه متحرک به‌عنوان یک بازی غیر همکارانه $G = (M, \{\Sigma_{m,t}\}, \{U_{m,t}\})$ روی شکاف‌های زمانی گسسته $t \in \{۱, ۲, \dots\}$ مدل‌سازی شده است [۷][۱۲].

بازیکنان: بازیکنان در بازی مجموعه $M = \{۱، ۲، \dots، M\}$ از گره‌های لبه موبایل هستند. هر گره به‌عنوان یک تصمیم‌گیرنده منطقی باهدف به حداکثر رساندن سود خود عمل می‌کند. در هر شکاف زمانی t، گره $m \in M$ فقط اطلاعات محلی در مورد منابع خود $(F_{m,t}, B_{m,t})$ و وظایف در ناحیه تحت پوشش خود را دارد و وضعیت شبکه جهانی را نمی شناسد.

استراتژی‌ها: گره m در هر نوبت زمانی t می‌تواند تصمیمات مختلفی در مورد تخصیص منابع خود بگیرد که هر کدام شانس مختلفی برای انجام دارند. مجموعه همه این احتمالات تصمیم‌گیری برای تخصیص منابع، فضای استراتژی $S_{m,t}$ نام دارد که نشان‌دهنده گزینه‌های ممکن برای گره m در زمان t است. یک استراتژی $\sigma_{m,t} = \{x_{n,m}, t\}$ مشخص می‌کند که کدام وظایف از دستگاه‌های همسایه برای پردازش پذیرفته می‌شوند، که در آن $x_{n,m,t} \in \{۰, ۱\}$.

$$\Sigma_{m,t} = \{\sigma_{m,t} |\, \sigma_{m,t}\ satisfies\ node\ m's\ capacity\ constraints\}$$

انتخاب راهبرد توسط ظرفیت‌های منابع گره برای اطمینان از امکان‌پذیر بودن تخصیص محدود می‌شود. راهبردها نیز تحت تأثیر بازده مورد انتظار از تخصیص به دستگاه‌های مختلف قرار می‌گیرند.

بهره‌وری: تابع ابزار $U_{m,t}(\sigma_{m,t}, \sigma_{-m,t})$ گره بازدهی دریافتی m را با گرفتن استراتژی $\sigma_{m,t}$ با توجه به استراتژی‌های $\sigma_{-m,t}$ از گره‌های دیگر دریافت می‌کند. $U_{m,t}$ را به‌صورت رابطه ۴ تعریف می‌کنیم:

$$U_{m,t}(\sigma_{m,t}, \sigma_{-m,t}) = \alpha U_{m,task}(\sigma_{m,t}) + (1-\alpha) U_{fairness}(\sigma_{m,t}, \sigma_{-m,t}) \qquad (۴)$$

در جایی که $U_{m,t}$ نشان‌دهنده بازده پردازش وظیفه بر اساس بارهای کاری پذیرفته‌شده است و $U_{fairness,t}$ عادلانه بودن تخصیص بین دستگاه‌ها را نشان می‌دهد. همچنین $\alpha \in [۰, ۱]$ اهداف را وزن می‌کند.

جدول زمانی: بازی در چند شکاف زمانی ادامه دارد $t = ۱, ۲, \dots$ هر شکاف شامل گره‌هایی است که استراتژی‌ها را به‌طور هم‌زمان بر اساس اطلاعات محلی برای توزیع وظایف انتخاب می‌کنند. استراتژی‌ها به‌صورت پویا با حجم وظیفه و تغییرات تحرک بین بازه‌ها سازگار می‌شوند.

تعادل: مجموعه استراتژی $\sigma^* = \{\sigma_1^*، \sigma_2^*، \dots\}$ یک تعادل نش را تشکیل می‌دهد، اگر هیچ یک از گره‌ها نتوانند با تغییر یک‌جانبه استراتژی خود، سود خویش را – با توجه به استراتژی‌های سایر گره‌ها – بهبود ببخشند. به‌این‌ترتیب، این مجموعه استراتژی‌ها تخصیص منابع را به‌صورت توزیع‌شده ثابت و پایدار می‌کند. به‌عبارت‌دیگر، در این وضعیت تعادلی هیچ گره‌ای منفعت بیشتری نخواهد یافت که موجب تغییر رفتار (استراتژی) خود شود.

۳-۳- مدل مدیریت منابع

ما مسئله تخصیص منابع توزیع‌شده را به‌عنوان یک بهینه‌سازی جهانی برای به حداکثر رساندن کل ابزار سیستم تحت محدودیت ظرفیت و تأخیر فرموله می‌کنیم که با توجه به ویژگی‌های شبکه و ابزارهای مورد استفاده، پارامترهایی مانند تاخیر میانگین و ظرفیت پهنای باند قابل تخمین، به‌همراه

سربار وارده از طرف شبکه قابل محاسبه است [۱۳][۱۴]. هدف، به حداکثر رساندن ابزار جمع‌آوری‌شده $\sum_{n\in N} U_n$ از همه وظایفی است که در دستگاه‌های $n \in N$ انجام می‌شوند.

$$U_n = \alpha_n * T_n \quad (۵)$$

جایی که T_n تعداد وظایفی است که توسط گره m انجام گرفته/پردازش شده است و α_n هم وزنی است که به گره m اختصاص داده می‌شود.

هدف این کار بهینه‌سازی توان عملیاتی و انصاف بین دستگاه‌ها است. متغیر تصمیم تخصیص عبارت است از:

$$x_{n,m,t} = \{۰,۱\}$$

که نشان می‌دهد که آیا وظیفه n در زمان t به گره m اختصاص داده‌شده است یا خیر. محدودیت‌ها (از اطلاعات تاریخی (گذشته) ذخیره شده برای محاسبه محدودیت‌ها استفاده شده است):

- ظرفیت CPU گره: $\sum_{m\in M} x_{n,m,t}C_{n,t} \le F_{m,t}$. اطمینان حاصل می‌کند که بار CPU از ظرفیت گره $F_{m,t}$، m تجاوز نمی‌کند.
- ظرفیت پهنای باند گره: $\sum_{n\in N} x_{n,m,t}B_{n,t} \le B_{m,t}$. در محدوده پهنای باند گره m : $B_{m,t}$.
- مهلت تأخیر: تأخیر $(n,t) \le L_{n,t}$. تأخیر همه وظیفه‌ها با مهلت $L_{n,t}$ مطابقت دارد.

این فرمول مشکل بهینه‌سازی جهانی زمان‌بندی وظایف به گره‌ها را با رعایت محدودیت‌های سیستم مدل‌سازی می‌کند. الگوریتم مبتنی بر walk Lévy (زیر بخش ۱-۳-) ما یک رویکرد توزیع‌شده برای تقریب این راه‌حل ارائه می‌دهد.

۴- روش پیشنهادی

ما الگوریتم توزیع‌شده‌ای (الگوریتم ۱) را طراحی کرده‌ایم که تعاملات مستقلی بین گره‌های لبه متحرک برای مذاکره و تخصیص منابع در نظر می‌گیرد. این الگوریتم به هر گره اجازه می‌دهد بر اساس الگوهای تحرک walk Lévy که الهام گرفته از طبیعت است، توزیع غیرمتمرکز و به‌صورت مستقل حرکت کند (شکل ۱-۳-۳) و با همسایگانش در مورد تخصیص منابع مذاکره و توافق نماید. در هر شکاف زمانی t، هر گره m مراحل زیر را برای تخصیص منابع از طریق چانه‌زنی محلی انجام می‌دهد:

به‌روزرسانی تحرک: گره m به‌طور تصادفی اندازه گام $l_{m,t}\sim L(\lambda)$ و جهت $\theta_{m,t}$ را برای به‌روزرسانی مکان خود انتخاب می‌کند. **کشف همسایه**: گره m گره‌های همسایه $N_{m,t}$ را که در محدوده ارتباطی آن قرار دارند، شناسایی می‌کند، که به طور میانگین ۲ الی ۴ میلی‌ثانیه باتوجه‌به تعداد همسایه‌ها زمان

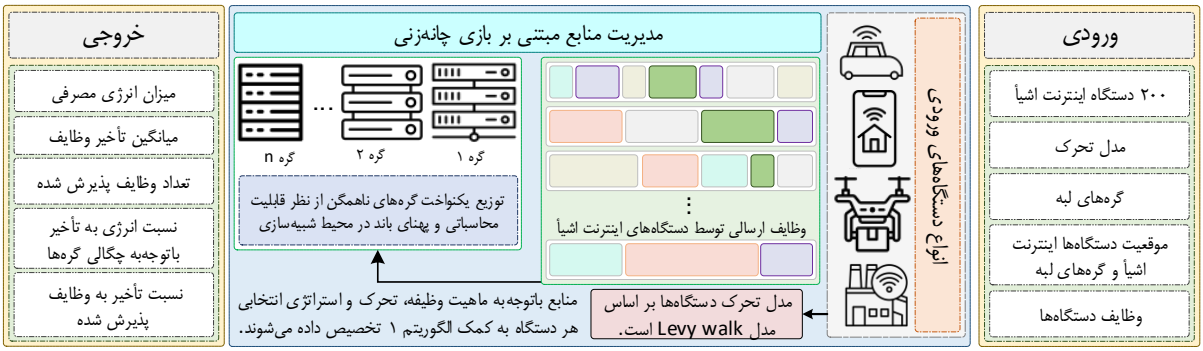
Algorithm 1: DR2A

Input: A set of Nodes with their location, Strategies of the game, Task requests received by nodes, Resources of nodes
Output: Bargaining strategy of nodes at time *t*, Finalized resource allocation, Processed tasks

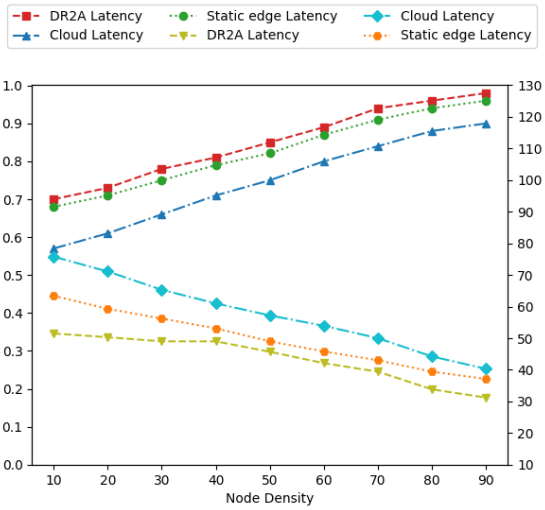
```
foreach time slot t do
    foreach node m do
        Draw step size l_{m,t}~L(λ)
        Draw direction θ_{m,t}~U(0, 2π)
        Update location x_{m,t+1}, y_{m,t+1}
        N_{m,t} ← Find nodes within range of x_{m,t+1}, y_{m,t+1}
        Receive tasks requests T_{m,t} from devices in coverage
        σ_{m,t} ← Compute initial bargaining strategy
        While (not all offers processed) do
            foreach node n ∈ N_{m,t} do
                p ← Generate provisional offer proportionate to resources
                Response ← Request response from n
                if (Response == Accept) then
                    S'_{m,t} ← Add (n, allocation) to schedule
                else
                    σ_{m,t} ← Update strategy using binary exponential backoff
        if (S'_{m,t} is feasible) then
            S_{m,t} ← Commit provisionally accepted offers
        else
            σ_{m,t} ← Best response strategy
            Go to bargaining step
        Process tasks allocated in S_{m,t}
```

می‌برد. **پذیرش وظیفه**: گره m درخواست‌های وظیفه را از دستگاه‌های موجود در ناحیه تحت پوشش جدید خود دریافت می‌کند. **چانه‌زنی**: گره m برای تخصیص منابع درگیر چانه‌زنی توزیع‌شده با همسایگان $N_{m,t}$ می‌شود. **تخصیص**: گره m تخصیص‌های مورد مذاکره با همسایگان را قطعی و نهایی می‌سازد و سپس وظایف پذیرفته‌شده را قبل از انقضای زمان t پردازش می‌کند. اکنون به جزئیات فرآیند چانه‌زنی و تخصیص در هر شکاف می‌پردازیم.

چانه‌زنی: گره m استراتژی پیشنهادی خود $\sigma_{m,t}$ را برای به حداکثر رساندن مطلوبیت $U_{m,t}$ محاسبه می‌کند. با استفاده از طرح عقب‌نشینی نمایی باینری، پیشنهادهای موقتی را به همسایگان متناسب با منابع موجود می‌دهد. همسایه‌ها با پذیرش/رد پاسخ می‌دهند تا زمانی که ظرفیت‌تکمیل شود یا همه پیشنهادها پردازش شود. این پویایی چانه‌زنی توزیع‌شده واقع‌بینانه را تقلید می‌کند. **تخصیص**: گره m پیشنهادها موقت پذیرفته‌شده را در برنامه



شکل ۱-۳-۳ نشان دهنده کلیت روش پیشنهادی مبتنی بر سیستم مدل



شکل ۲-۳-۳ نسبت‌پذیرش و میانگین تاخیر وظیفه به تعداد گره

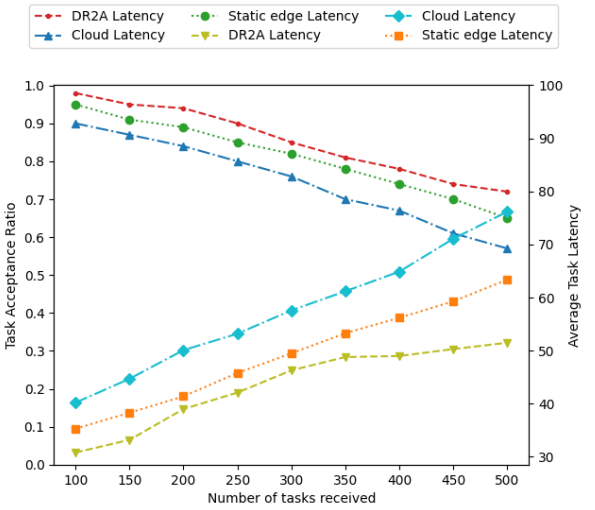
زمان‌بندی $S'_{m,t}$ خود ثبت می‌کند. اگر $S'_{m,t}$ در t امکان‌پذیر باقی بماند، متعهد است. در غیر این صورت، m با بهترین استراتژی واکنش $\sigma'_{m,t}$ دوباره چانه‌زنی می‌کند. این به‌طور مکرر با مذاکرات ناهمزمان ادامه می‌یابد تا زمانی که یک برنامه باثبات $S'_{m,t}$ در عرض t به دست آید. **همگرایی**: چانه‌زنی تضمین‌شده است که به تخصیص تعادل نش همگرا می‌شود، زیرا گره‌ها ابزارهای محلی را از طریق به‌روزرسانی استراتژی نزدیک‌بینی در طول زمان به حداکثر می‌رساند.

الگوریتم D2RA که دارای حلقه هایی بر روی گره‌ها و شکاف‌های زمانی است، پیچیدگی زمانی تا حد زیادی تحت تاثیر این دو عامل قرار می‌گیرد. اگر گره‌های n و شکاف‌های زمانی t وجود داشته باشد، و الگوریتم هر گره را در یک شکاف زمانی جداگانه پردازش کند، ممکن است با پیچیدگی زمانی پایه $O(n \times t)$ شروع کنیم.

۵- نتایج

ما یک شبیه‌سازی رویداد گسسته ایجاد می‌کنیم تا یک سیستم محاسباتی لبه موبایل توزیع‌شده جغرافیایی را در بازه‌های زمانی مدل‌سازی کنیم. شبیه‌ساز پارامترهای مختلفی مانند تعداد سرورهای لبه، نرخ رسیدن وظایف و ظرفیت پردازش هر سرور را در نظر می‌گیرد. منطقه شبیه‌سازی ۱۰۰۰ متر در ۱۰۰۰ متر با ۹۰ گره لبه متحرک ناهمگن ازنظر قابلیت‌های محاسباتی (۱۰-۱۰۰ گیگاهرتز) و شبکه (۱۰۰-۱۰۰۰ مگابیت بر ثانیه) است. تحرک آن‌ها از مدل Lévy walk پیروی می‌کند. تعداد ۲۰۰ دستگاه اینترنت اشیا وجود دارد که به‌طور یکنواخت در این منطقه توزیع‌شده‌اند که وظایف مهم تأخیر را در طول زمان ایجاد می‌کنند. ما شدت‌های مختلف بارکاری را شبیه‌سازی می‌کنیم: سبک (۵-۱۰ وظیفه در دقیقه/دستگاه)، متوسط (۱۰-۱۵ وظیفه در دقیقه/دستگاه) و سنگین (۱۵-۲۰ وظیفه در دقیقه/دستگاه). ما معیارهای کلیدی را برای مقایسه کارایی الگوریتم خودارزیابی می‌کنیم:

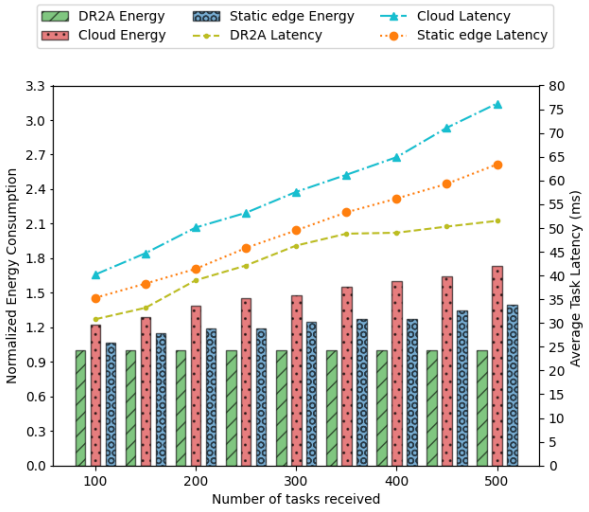
نسبت پذیرش وظیفه: نسبت وظایف پذیرفته‌شده به‌کل وظایف دریافتی در لبه، منعکس‌کننده استفاده از منابع است. **میانگین تأخیر وظیفه**: میانگین زمان لازم برای تکمیل اجرای وظایف پذیرفته‌شده. تأخیر کمتر انطباق



شکل ۴-۳-۳ نسبت‌پذیرش و میانگین تاخیر وظیفه به تعداد وظایف دریافتی توسط گره‌ها

توافقنامه سطح خدمات[1] را تضمین می‌کند. **بهره‌وری انرژی**: کل انرژی مصرف‌شده برای محاسبات و انتقال در هر وظیفه بهینه‌سازی مصرف انرژی برای MEC ضروری است. نتایج با طرح‌های پایه محک می‌شوند: بارگذاری ابری: همه وظایف در یک ابر متمرکز و راه دور بارگذاری می‌شوند. لبه ایستا: وظایفی که بر اساس مجاورت به گره‌های لبه غیر متحرک ثابت اختصاص داده می‌شود. **تخصیص Lévy walk**: وظایف به‌طور تصادفی به گره‌هایی که تحت تحرک Lévy walk هستند، اختصاص داده می‌شوند.

در شکل ۳-۳- نسبت پذیرش وظایف توسط DR2A (نقاط مربعی شکل با رنگ قرمز) نسبت به دو حالت دیگر با افزایش تعداد گره‌ها حداقل ۳٪ بهتر عملکرد دارد و میزان پذیرش وظایف تا نزدیک به ۱۰۰درصد نیز می‌رود. البته با افزایش تعداد گره‌ها میزان تأخیر نیز کاهش می‌یابد که وقتی ما تحرک را هم مدنظر می‌گیریم، می‌بینیم که همچنان الگوریتم ما (نقاط مثلثی شکل با رنگ زیتونی) به نسبت سایر مدل‌ها بهبود ۴٪ را دارا است. همچنین وظایف



شکل ۳-۳-۳ نسبت میزان مصرف انرژی و میانگین تاخیر وظایف به تعداد وظایف دریافتی توسط گره‌ها

[1] Service Level Agreement (SLA)

استفاده‌شده در شبیه‌سازی، دارای توزیع یکنواخت از سه مدل بارکاری است که نشان‌دهنده رعایت اعتدال بین هر سه مدل بارکاری هستیم که در شکل شماره ۳-۳-۳ مصرف انرژی و میانگین تأخیر برای گره‌ها را نسبت به تعداد وظایف دریافتی نشان می‌دهیم. در این نمودار مقدار مصرف انرژی الگوریتم DR2A در حالتی که گره‌ها دارای تحرک هستند نسبت به دو مدل دیگر که برای گره‌ها مدل نظر گرفته‌شده است حداقل ۷٪ و حداکثر ۲۲٪ بهتر عمل کرده است، چراکه الگوریتم توزیع‌شده بر اساس مدل تحرک گره‌ها وظایف را به آن‌ها اختصاص می‌دهد. از طرف دیگر با افزایش تعداد وظایف دریافتی میزان تأخیر در مدل ابری و لبه‌های ایستا ۸٪ و ۶٪ بدتر از روش DR2A است. در شکل ۴-۳-۳ مقدار تأخیر به نسبت تعداد وظایف پذیرش‌شده از جانب گره‌ها را نشان می‌دهد که درواقع مقایسه‌ای در رابطه با افزایش تعداد وظایف تأثیر تأخیر در پذیرش وظایف به چه صورت است که مشاهده می‌شود با افزایش تعداد وظایف، میزان تأخیر نیز افزایش می‌یابد که این باعث کاهش تعداد وظایف دریافتی توسط گره‌ها خواهد شد و البته تعداد وظایف پذیرش‌شده با الگوریتم DR2A بیشتر از ۶٪ سایر حالات که در نمودار ۴-۳-۳ نمایان است و این به خاطر ماهیت تحرکی است که در الگوریتم مدنظر گرفته‌شده است.

۶- جمع‌بندی

در این کار، ما یک الگوریتم توزیع‌شده مبتنی بر نظریه بازی برای مدیریت منابع مشارکتی در سامانه‌های محاسباتی لبه تلفن همراه که از برنامه‌های اینترنت اشیا بی‌درنگ پشتیبانی می‌کنند، پیشنهاد کردیم. رویکرد ما با بهره‌گیری از بینش‌های رفتارهای جستجوی طبیعی، مبتنی بر حرکت گره‌ها در Lévy walk و تخصیص وظایف از طریق چانه‌زنی محلی است. شبیه‌سازی‌های گسترده، رویکرد را در میان شدت‌های بارکاری و تراکم شبکه‌های مختلف ارزیابی کردند. نتایج بهبود نسبت پذیرش وظیفه ۳٪، کاهش تأخیر ۶٪ و صرفه‌جویی در انرژی ۲۲٪ در مقایسه با تخلیه ابری و لبه ایستا. این کارایی مهار پویایی همکاری لبه‌های اضطراری را از الگوهای تحرک Lévy walk نشان می‌دهد. نشان داده شد که الگوریتم توزیع‌شده به‌طور مستقل استفاده از منابع را به حداکثر می‌رساند و معیارهای برنامه را در یک محیط پویا بهینه می‌کند. به‌عنوان بخشی از کار آینده، ما قصد داریم تا سناریوهای پیچیده‌تر را با گره‌های متنوع و تحرک غیر Lévy تجزیه‌وتحلیل کنیم. توسعه پروتکل‌های چانه‌زنی سبک‌وزن و مکانیسم‌های تشویقی برای مشارکت نیز می‌تواند عملی بودن راه‌حل را بهبود بخشد. ادغام قابلیت اطمینان و حفاظت‌های امنیتی، قابلیت استقرار چارچوب را در تنظیمات حیاتی اینترنت اشیا افزایش می‌دهد.

مراجع

[۱] L. A. Haibeh, M. C. E. Yagoub and A. Jarray, "A Survey on Mobile Edge Computing Infrastructure: Design, Resource Management, and Optimization Approaches," in *IEEE Access*, vol. 10, pp. 27591-27610, 2022.

[۲] S. Douch, M. R. Abid, K. Zine-Dine, D. Bouzidi and D. Benhaddou, "Edge Computing Technology Enablers: A Systematic Lecture Study," in *IEEE Access*, vol. 10, pp. 69264-69302, 2022.

[۳] F. Al-Doghman, N. Moustafa, I. Khalil, N. Sohrabi, Z. Tari and A. Y. Zomaya, "AI-Enabled Secure Microservices in Edge Computing: Opportunities and Challenges," in *IEEE TSC*, vol. 16, no. 2, pp. 1485-1504, 1 March-April 2023.

[۴] S. Yang, Z. Su, Q. Xu, R. Xing and D. Fang, "Task Allocation Optimization Strategy in UAV-enabled Mobile Edge Computing System," 2023 *IEEE (MetaCom)*, Kyoto, Japan, 2023, pp. 338-344.

[۵] J. Moura and D. Hutchison, "Game Theory for Multi-Access Edge Computing: Survey, Use Cases, and Future Trends," in *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 260-288, Firstquarter 2019.

[۶] S. Arisdakessian, O. A. Wahab, A. Mourad, H. Otrok and N. Kara, "FoGMatch: An Intelligent Multi-Criteria IoT-Fog Scheduling Approach Using Game Theory," in *IEEE/ACM TN*, vol. 28, no. 4, pp. 1779-1789, Aug. 2020.

[۷] H. Lu, G. Xu, C. W. Sung, S. Mostafa and Y. Wu, "A Game Theoretical Balancing Approach for Offloaded Tasks in Edge Datacenters," 2022 *IEEE 42nd Int. Conf. Dist. Com. Sys. (ICDCS)*, Bologna, Italy, 2022, pp. 526-536.

[۸] Liqing Liu, Zheng Chang, Xijuan Guo and T. Ristaniemi, "Multi-objective optimization for computation offloading in mobile-edge computing," 2017(ISCC), Heraklion, Greece, 2017, pp. 832-837.

[۹] M. A. Hassan, M. Xiao, Q. Wei, and S. Chen, ‘‘Help your mobile applications with fog computing,’’ in *Proc. 12th Annu. IEEE Int. Conf. Sens., Commun., Netw.-Workshops (SECON Workshops)*, Jun. 2015, pp. 1–6

[۱۰] Rukmani Panjanathan and Ganesan Ramachandran, "Enhanced low latency queuing algorithm with active queue management for multimedia applications in wireless networks", *International Journal of High Performance Computing and Networking*, vol. 10, no. 1-2, pp. 23-33, 2017

[۱۱] Jie Zhang, Guangjie Han and Yujie Qian, "Queuing theory based co-channel interference analysis approach for high-density wireless local area networks", *Sensors*, vol. 16, no. 9, 2016.

[۱۲] S. Kim, "Bargaining Game Based Offloading Service Algorithm for Edge-Assisted Distributed Computing Model," in *IEEE Access*, vol. 10, pp. 63648-63657, 2022.

[۱۳] W. Duan, X. Gu, M. Wen, Y. Ji, J. Ge and G. Zhang, "Resource Management for Intelligent Vehicular Edge Computing Networks," in *IEEE TITS*, vol. 23, no. 7, pp. 9797-9808, July 2022.

[۱۴] T. Bahreini, M. Brocanelli and D. Grosu, "VECMAN: A Framework for Energy-Aware Resource Management in Vehicular Edge Computing Systems," in *IEEE TMC*, vol. 22, no. 2, pp. 1231-1245, 1 Feb. 2023.

پنجمین کنفرانس ملی انفورماتیک ایران
پژوهشگاه دانشهای بنیادی، پردیس فرمانیه، تهران
۱۳ و ۱۴ دی ماه ۱۴۰۲

یک چارچوب انتخاب ویژگی مرکب مبتنی بر معیار حداقل افزونگی و حداکثر ارتباط برای طبقه بندی داده‌های بیولوژیکی

فاطمه کوکب‌زاده[1]، الهام عباسی هرفته[2] ، جمال زارعپور احمدآبادی[3]

[1] بخش علوم کامپیوتر، دانشکده علوم ریاضی، دانشگاه یزد، یزد
Kokabzadehfatemeh@gmail.com
[2] بخش علوم کامپیوتر، دانشکده علوم ریاضی، دانشگاه یزد، یزد
E.abbasi@yazd.ac.ir
[3] بخش علوم کامپیوتر، دانشکده علوم ریاضی، دانشگاه یزد، یزد
Zarepourjamal@yazd.ac.ir

چکیده

انتخاب ویژگی به مشکل یافتن زیرمجموعه بهینه ویژگی‌ها با حذف ویژگی های نامربوط و زائد برای بهبود دقت طبقه‌بندی اشاره دارد. در این مقاله یک چارچوب انتخاب ویژگی مرکب مبتنی بر روش های نظارت شده و بدون نظارت بر اساس فاصله با استفاده از ایده معیارهای حداقل افزونگی و حداکثر ارتباط برای طبقه بندی داده های بیولوژیکی به کار گرفته شده است.

این مطالعه به بررسی و مقایسه عملکرد روش‌های انتخاب ویژگی می‌پردازد. ویژگی‌های دارای بالاترین رتبه با استفاده از یک آستانه تجربی انتخاب می شوند. برای ارزیابی ویژگی‌های انتخاب‌شده، دو طبقه‌بندی‌کننده، یعنی درخت تصمیم و ماشین بردار پشتیبان بر روی مجموعه داده‌های باینری که اتخاذ شده از مخزن داده UCI ، اعمال شده است. نتایج تجربی برتری روش ها را از نظر بهبود دقت طبقه‌بندی نشان می‌دهد.

کلمات کلیدی

انتخاب ویژگی، انتخاب ویژگی بدون‌نظارت، انتخاب ویژگی نظارت‌شده، اندازه‌گیری فاصله.

1- مقدمه

انتخاب ویژگی یک روش موثر است که هدف آن دست‌یابی به یک زیرمجموعه کوچک از ویژگی‌های انتخاب شده از فضای ویژگی اصلی با حذف ویژگی‌های نامربوط یا اضافی است. با کاهش ویژگی، تفسیر مدل‌های یادگیری راحت‌تر و در زمان محاسباتی صرفه‌جویی می‌شود. بسته به قابلیت دسترسی برچسب ویژگی‌ها، روش‌های انتخاب ویژگی به دو دسته‌ی نظارت‌شده و بدون نظارت گروه‌بندی می‌شوند. روش انتخاب ویژگی نظارت‌شده باتوجه به کلاس نمونه‌ها، ویژگی‌های متمایزکننده را برای طبقه‌بندی انتخاب می‌کند. روش انتخاب ویژگی بدون‌نظارت، ویژگی‌های با اهمیت کمتر را فیلتر می‌نماید. بر اساس معیار ارزیابی و چگونگی ترکیب با الگوریتم‌های طبقه‌بندی، روش‌های انتخاب ویژگی نظارت‌شده و بدون نظارت به سه کلاس فیلتر[1]، بسته‌بندی[2] و تعبیه‌شده[3] گروه‌بندی می‌شوند [3,6,12,14,19].

روش‌های فیلتر، مهم‌ترین ویژگی‌ها را مستقل از طبقه‌بندی‌کننده با استفاده از مشخصه‌های آماری داده‌ها انتخاب می‌کند. این روش‌ها معمولاً شامل دو مرحله هستند: (1) رتبه‌بندی همه ویژگی‌ها بر اساس یک معیار خاص، (2) انتخاب ویژگی‌های دارای بالاترین رتبه. یکی از معایب روش‌های فیلتر این است که ویژگی‌ها را بر اساس همبستگی بین آنها متمایز نمی‌کند. در نتیجه، ویژگی‌های انتخاب شده ممکن است اضافی باشند. روش‌های بسته‌بندی، طبقه‌بندی‌کننده را به‌روی زیرمجموعه‌های مختلف ویژگی‌ها اعمال می‌کنند و بهترین زیرمجموعه ویژگی را بر اساس معیارهای عملکرد طبقه‌بندی‌کننده مانند یادآوری (حساسیت) و دقت انتخاب می‌نمایند. یکی از معایب روش‌های بسته‌بندی ،حجم محاسباتی زیاد و سربار جستجو می‌باشد.

روش‌های فیلتر از نظر محاسباتی نسبت به روش‌های بسته بندی کارایی بهتری دارند و با مجموعه داده‌های مختلف سازگار هستند، که آنها را برای مسائل با ابعاد بالا مناسب‌تر می‌نماید. برای ارزیابی ویژگی‌های انتخاب‌شده، سه طبقه‌بندی‌کننده، یعنی درخت تصمیم[4]، ماشین بردار پشتیبان[5] و ساده بیز[6] در پژوهش گذشته [4] در این زمینه درنظرگرفته‌شده است. یکی از معیارهای ارزیابی روش‌های انتخاب ویژگی اندازه گیری ارتباط[7] و افزونگی[8] می‌باشد. از معیارهای مختلفی نظیر فاصله و همبستگی برای اندازه‌گیری

---

[1] Filter
[2] Wrapper
[3] Embedded
[4] Decision Tree
[5] Support Vector Machine
[6] Naive Bayes
[7] Relevance
[8] Redundancy

ارتباط و افزونگی استفاده می‌شود. به‌عنوان مثال روش حداقل – افزونگی – حداکثر –ارتباط[9] (mRMR) [13] یک اندازه گیری ارتباط و افزونگی بر اساس فاصله برای انتخاب ویژگی در یک فرآیند بدون‌نظارت می‌باشد.

در رویکرد mRMR ارائه شده توسط پنگ و همکاران [13] از اطلاعات دوطرفه (MI)[10] برای به‌حداکثررساندن ارتباط ویژگی‌ها با کلاس نمونه‌ها و به‌حداقل‌رساندن افزونگی در بین ویژگی‌های انتخاب‌شده استفاده می‌شود. اگرچه روش مذکور به‌صورت کلی مؤثر است، اما زمان و حجم محاسباتی آن زیاد است. آن‌ها از تفاوت همبستگی (FCD)[11] برای محاسبه ارتباط و برای ارزیابی افزونگی از معیارهای دیگر مانند فاصله اقلیدسی و ضریب همبستگی پیرسون استفاده کردند. در [15] ویژگی‌ها به‌صورت نزولی بر اساس ارتباط آنها با برچسب‌های کلاس رتبه‌بندی شده است، درجه ارتباط بین دو ویژگی با استفاده از عدم قطعیت متقارن (SU)[12] تخمین و از حداکثر نرخ افزونگی (MRR)[13] استفاده شده است و رتبه‌بندی ویژگی به‌عنوان آستانه افزونگی به‌دست آمده است. یک انتخاب ویژگی مبتنی بر وابستگی برای یادگیری بدون نظارت توسط لیم و کیم در [9] طراحی شد که اطلاعات دوطرفه بین دو ویژگی را اندازه‌گیری می‌کند و برای جلوگیری از افزونگی، ویژگی‌های مستقل را اولویت می‌دهد. همچنین بررسی مبادله بین ارتباط و افزونگی یک زمینه تحقیقاتی فعال در انتخاب ویژگی با هدف بهبود کیفیت زیرمجموعه ویژگی انتخاب شده است. آگاروال و گوپتا در [1] رویکرد mRMR به‌کارگرفته شده در [7]را به همراه دو رویکرد پیشرفته به‌کار بردند و مجموعه‌ای از ویژگی‌های 8 بعدی را پیشنهاد کردند. بوگاتا و دروتار در [16] شکل دیگری از هم‌ارزی را برای تعمیم mRMR ارائه کردند و هم‌چنین از معیارهای وابستگی مانند ضریب همبستگی پیرسون، آمار آزمون کای اسکوئر، ضریب همبستگی اسپیرمن و همبستگی فاصله علاوه بر اطلاعات دوطرفه استفاده کردند. عابدی در [2] برای روش اولویت سفارش بر اساس شباهت به راه حل ایده آل[14] (TOPSIS) 15 معیار فاصله را بررسی کرد. ناث در [11] از معیارهای فاصله متفاوتی مانند ماتوزیتا [15]، تانیموتو [16]و غیره در الگوریتم‌های خوشه‌بندی برای نشان دادن تعداد نویسندگان یک سند استفاده کرد. پراساتا و همکاران در [15] اندازه‌گیری‌های فاصله را از هشت خانواده اصلی فاصله ارزیابی کردند تا دریابند که کدام‌یک به‌عنوان هسته طبقه‌بندی K – نزدیکترین همسایه بهترین عملکرد را دارد. کوچر و ساوی در [10] به برتری بالاتر معیارهای تانیموتو، ماتوزیتا و کلارک[17] در مقایسه با معیار کسینوس در ارزیابی شباهت یا فاصله بین اسناد در خوشه‌بندی تعیین پروفایل نویسنده پی‌بردند. در [5] ایکای، ناث و یک تابع K – نزدیکترین همسایه[18] را بر اساس فاصله منهتن در یک روش انتخاب ویژگی اعمال کردند. ارگن، کومرت، توگاکار در [8] الگوریتم mRMR را همراه با طبقه‌بندی‌کننده K – نزدیکترین همسایه با استفاده از فاصله اقلیدسی پیشنهاد کردند. در [18] وانگ و لی یک روش فیلتر را با استفاده از معیار اقلیدسی بر اساس به حداکثر رساندن فاصله بین کلاسی و درعین‌حال به‌حداقل‌رساندن فاصله درون کلاسی پیشنهاد کردند.

2- روش پیشنهادی

در این مقاله روش پیشنهادی در ادامه پژوهش انجام گرفته در [4] می‌باشد. در ادامه به بیان جزئیات روش ارائه‌شده در [4] خواهیم پرداخت و ایده پیشنهادی بیان خواهدشد.

فرض کنیم مجموعه داده استاندارد با ابعاد بالا در قالب یک ماتریس، دارای ابعاد $n \times d$ با n نمونه و d ویژگی به‌صورت زیر است:

$$X = \begin{pmatrix} x_1 \\ x_2 \\ . \\ . \\ . \\ x_n \end{pmatrix} = \begin{pmatrix} f_{11} & f_{12} & \cdots & f_{1d} \\ f_{21} & f_{22} & \cdots & f_{2d} \\ . & . & . & . \\ . & . & . & . \\ . & . & . & . \\ f_{n1} & f_{n2} & \cdots & f_{nd} \end{pmatrix}$$

از x_1 تا x_n هر x_i نشان‌دهنده ردیف iام ماتریس X است و به‌عنوان یک بردار d بعدی در نظر گرفته می‌شود که به‌صورت زیر است:

$$\overline{x_i} = [f_{i1}, f_{i2}, \ldots, f_{id}]; i = 1,2,\ldots,n$$

F مجموعه‌ای از کل ویژگی‌های X است. $(F = [F_1, F_2, \cdots, F_d])$ که هر F_j نشان‌دهنده ستون jام ماتریس X است و به‌عنوان یک بردار n بعدی در نظر گرفته می‌شود که به‌صورت زیر است:

$$\overline{F_j} = [f_{1j}, f_{2j}, \ldots, f_{nj}]^T; j = 1,2,\ldots,d$$

در انتخاب ویژگی نظارت‌شده بر اساس روش mRMR ، ویژگی‌ها بر اساس معادله 2 رتبه بندی می‌شوند. به‌عنوان مثال، در یک مجموعه داده زیست پزشکی که یک مشکل دودویی را نشان می‌دهد، یک کلاس به افراد بیمار و کلاس دیگر به افراد سالم اختصاص داده می‌شود که به‌ترتیب با علائم $+$ و $-$ نشان داده می‌شوند. بنابراین F به دو قسمت مجزا تقسیم می‌شود:

$$\alpha_{F_j} = dist\left(\overline{F_j^+}, \overline{F_j^-}\right) \qquad (2)$$

فاصله بین هرنمونه بیمار و همه‌ی نمونه‌های سالم را محاسبه می‌کند که

$$\overline{F_j^+} = \left(f_{1j}^+, f_{2j}^+, \ldots, f_{s,j}^+\right)^t$$

کلاس افراد بیمار و

$$\overline{F_j^-} = \left(f_{1j}^-, f_{2j}^-, \ldots, f_{s,j}^-\right)^t$$

کلاس افراد سالم است.

ابتدا، در هر بردار ویژگی، فواصل بین نمونه‌های کلاس‌های مختلف به‌عنوان رتبه ویژگی ذخیره می‌شود (در مثال بالا، فاصله بین هر نمونه بیمار و همه نمونه‌های سالم). سپس، رتبه‌های ویژگی‌ها به ترتیب نزولی مرتب می‌شوند. درنهایت تعداد) tکه به‌صورت تجربی از پیش تعریف‌شده) از ویژگی‌های بالاترین رتبه انتخاب می‌شود. ویژگی‌ها با بالاترین رتبه نشان‌دهنده بزرگ‌ترین فاصله بین نمونه‌های آن‌ها از کلاس‌های مختلف است. بنابراین، ویژگی‌هایی

[9] Minimal-Redundancy -Maximal-Relevance
[10] Mutual-Information
[11] F- test-Correlation-Difference
[12] Symmetrical-Uncertainty
[13] Maximum-Rate-Redundancy
[14] Technique for-Order-Preference by-Similarity to Ideal -Solution
[15] Matusita
[16] Tanimoto
[17] Clark
[18] KNN

که باعث ایجاد چنین تبعیضی در فضای نمونه می‌شوند، نماینده‌ی مناسبی برای ویژگی‌های اصلی می‌باشند.

برای روش ترکیب، از روش رتبه‌بندی ترکیبی مبتنی بر شباهت، که برای تجمیع زیرمجموعه‌های ویژگی‌های به‌دست‌آمده از روش‌های انتخاب ویژگی استفاده می‌شود [20]، الگو گرفته تا بهترین زیرمجموعه از ویژگی‌ها انتخاب شود. رتبه مجموعه Rank-ensemble برای هر ویژگی F با استفاده از معادله 3 محاسبه می‌شود:

$$\text{Rank} - \text{ensemble} = \frac{\text{Rank-sup}}{n} \times \frac{\text{Rank-unsup}}{n} \quad (3)$$

که در آن Rank-sup و Rank-unsup به ترتیب رتبه نظارت‌شده و رتبه بدون‌نظارت به‌دست آمده برای هر ویژگی هستند و n نشان دهنده تعداد ویژگی‌های برتر انتخاب‌شده است. با توجه به روش رتبه‌بندی در نظر گرفته‌شده و مجموعه داده‌ها می‌توان نتایج متفاوت به دست‌آورد [4].

معیار به‌کارگرفته‌شده برای اندازه‌گیری ارتباط و افزونگی ویژگی‌های انتخابی در روش mRMR تأثیر قابل‌توجهی بر عملکرد آن دارد. در این پژوهش، به دلیل زمان اجرای زیاد الگوریتم بدون‌نظارت روش mRMR ، تاثیر ترکیب روش‌های بدون‌نظارت(SFI)[19] و شبح[20] و روش‌های نظارت‌شده مربع کای[21] و نظارت‌شده روش mRMR مورد بررسی قرار گرفت. هم‌چنین در روش ترکیب مبتنی بر شباهت تاثیر معیارهای اندازه‌گیری فاصله کانبرا، کلارک بررسی شد.

$$dist_{Clark}(A,B) = \sqrt{\sum_{i=1}^{m}\left(\frac{|a_i - b_i|}{a_i + b_i}\right)^2} \quad (4)$$

$$dist_{Canbera}(A,B) = \sum_{i=1}^{m}\frac{|a_i - b_i|}{a_i + b_i} \quad (5)$$

جدول (1) : مجموعه داده‌های بیولوژیکی در نظر گرفته شده در این مطالعه

Database name	Samples	Features
ALLAML	72	7129
leukemia	72	7070

جدول(2) : روش‌های در نظر گرفته شده در این مطالعه

Method name	Description	Name used
Specture	Unsupervised	Unsup1
Statistical_Feature_Importance	Unsupervised	Unsup2
Chisquare	Supervised	Sup1
mRMR	Supervised	Sup2
Ensemble1	Sup1 and Unsup1	Ens1
Ensemble2	Sup1 and Sup2	Ens2
Ensemble3	Unsup1 and Unsup2	Ens3

3- بحث

طبق جدول زیر برای داده‌ی ALLAML نتایج برخی از روش‌های ترکیب مانند Ens3 در DT بامعیار کانبرا با اختلاف 10 درصد بهتر از روش‌های منفرد نظارت‌شده‌ی Unsup1 و Unsup2 وبا اختلاف 2 الی 6 درصد از روش‌های بدون‌نظارت می‌باشد و همین طور در مقایسه با سایر روش‌های ترکیب Ens2 وEns1 ، 3 الی 7 درصد بهبود به‌دست‌آمده‌است. در معیار کلارک نیز روش‌های Ens3 وEns1 نسبت به روش‌های نظارت‌شده و بدون‌نظارت بهبود 1 الی 9 درصد رخ داده است که بیشترین بهبود روش‌های ترکیب درمقایسه با الگوریتم‌های بدون‌نظارت می‌باشد. در طبقه‌بندی‌کننده‌ی SVM در هر دو معیار اختلاف قابل توجهی در بین روش های ترکیب و منفرد دیده نشده است.

در همه‌ی روش‌ها طبقه‌بندی‌کننده‌ی SVM نتایج بهتری از DT ارائه کرده است.

جدول (3) : میانگین دقت روش‌های در نظر گرفته شده برای داده ALLAML در 10 بار آزمایش

Name used	Clark Distance Measure		Canberra Distance Measure	
	SVM	DT	SVM	DT
Unsup1	0.98	0.83	0.96	0.82
Unsup2	0.94	0.87	0.95	0.82
Sup1	0.99	0.84	0.97	0.90
Sup2	0.94	0.91	0.96	0.86
Ens1	0.97	0.92	0.98	0.85
Ens2	0.96	0.84	0.97	0.89
Ens3	0.96	0.91	0.97	0.92

طبق جدول زیر برای داده‌ی leukemia در معیار کانبرا و کلارک نتایج هیچ یک از روش‌های ترکیب بهتر از روش mRMR نظارت‌شده نمی‌باشد. اما میزان بهبود روش ترکیب Ens3 که ترکیب دو الگوریتم بدون‌نظارت است، نسبت به روش‌های بدون‌نظارت در هر دو طبقه‌بندی‌کننده از 4 الی 35 درصد است که بسیار قابل توجه می‌باشد، اما نسبت به روش‌های نظارت‌شده

[19] Statistical_Feature_Importance
[20] Specture
[21] Chisquare

بهبود قابل توجهی رخ نداده است. برعکس داده‌ی قبلی طبقه‌بندی‌کننده‌ی DT در اکثر روش ها نتایج بهتری از SVM ارائه کرده است.

جدول (4) : میانگین دقت روش‌های در نظر گرفته شده برای داده leukemia در 10 بار آزمایش

Name used	Clark Distance Measure		Canberra Distance Measure	
	SVM	DT	SVM	DT
Unsup1	0.59	0.90	0.88	0.69
Unsup2	0.79	0.93	0.90	0.78
Sup1	0.92	0.97	0.96	0.97
Sup2	0.94	0.99	0.99	0.94
Ens1	0.94	0.97	0.96	0.94
Ens2	0.90	0.97	0.96	0.92
Ens3	0.94	0.97	0.99	0.94

4- نتیجه گیری

در این مقاله، ابتدا کل مجموعه داده به صورت تصادفی به یک مجموعه آموزشی(80٪) و یک مجموعه آزمایشی(20٪) تقسیم می‌شود. سپس، مدل یادگیری بر روی داده های آموزشی ساخته می‌شود و بامحاسبه معیارهای عملکرد طبقه‌بندی بر روی مجموعه آزمون ارزیابی می‌شود.

روش‌های انتخاب ویژگی فیلتر براساس معیارهای اندازه‌گیری فاصله‌ی کلارک و کانبرا برای سناریوهای نظارت‌شده و بدون نظارت پیشنهاد شده‌اند. سپس ترکیب‌هایی از روش‌های نظارت شده و بدون نظارت برای انتخاب زیرمجموعه‌ای از ویژگی‌ها استفاده شد.

برای مطالعه اثربخشی روش‌های نظارت شده و بدون نظارت در حذف ویژگی‌های نامربوط، 10 بار روی هر مجموعه داده با درصد حذف 80٪ ویژگی های نامربوط از ویژگی‌های کل آزمایش شده است تا 20٪ از مرتبط ترین ویژگی ها را انتخاب کنند.

برای دو مجموعه داده‌ی مورد استفاده نتایج تجربی کمی دردسترس می‌باشد.

مراجع

[1] Agarwal, A., Gupta, A., *"A maximum relevancy and minimum redundancy feature selection approach for median filtering forensics"*, Multimedia Tools and Applications, vol. 79, pp. 21743-21770, 2020.

[2] Abedi, M., *"Non-Euclidean distance measures in spatial data decision analysis: investigations for mineral potential mapping"*, Annals of Operations Research, vol. 303, pp. 29-50, 2020.

[3] Anuradha, J., Venkatesh, B., *"A Review of Feature Selection and Its Methods"*, Cybernetics and information technologies, vol. 19, pp. 3-26, 2019.

[4] Akbari, E., Hallajian, B., Motameni, H., *"Ensemble feature selection using distance-based supervised and unsupervised methods in binary classification"* ,Expert Systems with Applications, vol. 200, p. 116794, 2022.

[5] Savoy, J., Ikae, C., Nath, S., *Conference and Labs of the Evaluation Forum :UniNE at PAN-CLEF 2019: Bots and Gender Task*, 2019.

[6] Zhao, Z., Liu, H., Motoda, H., Setiono, R., *Feature Selection : An Ever Evolving Frontier in Data Mining*, 2010.

[7] Torrecilla, L., Berrendero, J. R., Cuevas, A., *"The mRMR variable selection method: a comparative study for functional data"* , Journal of Statistical Computation and Simulation, vol. 86, pp. 891-907, 2015.

[8] Toğaçar, M., Ergen, B., Cömert, Z., *"Detection of lung cancer on chest CT images using minimum redundancy maximum relevance feature selection method with convolutional neural networks"*, Biocybernetics and Biomedical Engineering, vol. 40, pp. 23-39, 2020.

[9] Lim, H., Kim, D.-W., *"Pairwise dependence-based unsupervised feature selection"*, Pattern Recognit, vol. 111, p. 107663, 2021.

[10] Kocher, M., Savoy, J., *"Evaluation of text representation schemes and distance measures for authorship linking"*, Digit. Scholarsh. Humanit., vol. 34, pp. 189-207, 2019.

[11] Nath, S., *Conference and Labs of the Evaluation Forum :Style Change Detection by Threshold Based and Window Merge Clustering Methods*, 2019.

[12] Miao, J., Ping, Y., Chen, Z., Jin, X.-B., Li, P., Niu, L., *"Unsupervised feature selection by non-convex regularized self-representation"*, Expert Syst. Appl., vol. 173, p. 114643, 2021.

[13] Ding, C., Peng, H., *"Minimum redundancy feature selection from microarray gene expression data"*, Journal of bioinformatics and computational biology, vol. 3, pp. 185-205, 2005.

[14] Yu, Q., Jiang, S., Wang, R., yang Wang, H., *"A feature selection approach based on a similarity measure for software defect prediction"*, Frontiers of Information Technology & Electronic Engineering, vol. 18, pp. 1744-1753, 2017.

[15] Solorio-Fernández, S., Martínez-Trinidad, J. F., Carrasco-Ochoa, J. A., *"A Supervised Filter Feature Selection method for mixed data based on Spectral Feature Selection and Information-theory redundancy analysis"*, Pattern Recognit. Lett., vol. 138, pp. 321-328, 2020.

[16] Bugata, P., Drotár, P., *"On some aspects of minimum redundancy maximum relevance feature selection"*, Science China Information Sciences, vol. 63, 2019.

[17] Prasath, V. B. S., Alfeilat, H. A. A., Hassanat, A. B., Lasassmeh, O., Ahmad, S., Tarawneh, Alhasanat, M. B., Salman, H. E., *Effects of Distance Measure Choice on KNN Classifier Performance-A Review*, 2019.

[18] Wang, Y., Li, T., *"Local feature selection based on artificial immune system for classification"*, Appl. Soft Comput., vol. 87, p. 105989, 2020.

[19] Saeys, Y., Inza, I., Larrañaga, P., *"A review of feature selection techniques in bioinformatics"*, Bioinformatics, vol. 23 19, pp. 2507-17, 2007.

[20] Sadeghian, Z., Akbari, E., Nematzadeh, H., *"A hybrid feature selection method based on information theory and binary butterfly optimization algorithm"*, Eng. Appl. Artif. Intell., vol. 97, p. 104079, 2021.

پنجمین کنفرانس ملی انفورماتیک ایران
پژوهشگاه دانشهای بنیادی، پردیس فرمانیه، تهران
۱۳ و ۱۴ دی ماه ۱۴۰۲

پیش‌بینی بیماری‌های مزمن با داده‌های نامتوازن توسط ماشین بردار پشتیبان گرانشی

عبدالله محمدی[۱]، جلال‌الدین نصیری[۲] ، سهراب عفتی[۳]
[۱] دانشجوی کارشناسی ارشد علوم داده، ریاضیات کاربردی، دانشگاه فردوسی مشهد ، مشهد
abdhmohammadi@mail.um.ac.ir
[۲] استادیار، گروه ریاضی کاربردی، دانشکده ریاضی دانشگاه فردوسی مشهد، مشهد
jnasiri@um.ac.ir
[۳] استاد، گروه ریاضی کاربردی، دانشکده ریاضی دانشگاه فردوسی مشهد، مشهد
s-effati@um.ac.ir

چکیده

با پیشرفت تکنولوژی، روش‌های مبتنی بر داده برای تشخیص انواع بیماری‌ها، به طور گسترده‌ای مورد توجه قرار گرفته است. در طبقه بندی بیماری‌ها، تشخیص درست فرد "ناسالم" نسبت به تشخیص درست یک فرد سالم از اهمیت بیشتری برخودار است. اغلب داده‌های این بیماری‌ها دارای جامعه‌ی بیمار کوچک و جامعه‌ی سالم بزرگتری است. در این مقاله با تعریف یک تابع وزن ویژه در مدل وزنی الگوریتم twin svm[1] ، نشان داده می شود اختصاص وزن به گروه کوچکتر می تواند در تشخیص طبقه‌ی نمونه‌ها موثرتر باشد، ابتدا مفاهیم پایه‌ای مدل را بیان نموده سپس علاوه بر روال الگوریتم‌های دیگر، برای داده‌های کلاس کوچکتر نیز وزن اختصاص داده می شود. سپس از چندین مجموعه داده‌ی بیماری‌های مزمن مانند سرطان، دیابت و آلزایمر و ...، برای ارزیابی عملکرد روش استفاده نموده با مقایسه‌ی نتایج با چند روش دیگر، نشان داده می شود روش مورد استفاده می‌تواند با دقت بهتری نمونه‌ها را طبقه‌بندی کرده، نمونه‌های کلاس کوچکتر را نیز با دقت بالاتری تشخیص دهد بنابراین می‌توان انتظار داشت بتواند بر روش‌های دیگر برتری داشته باشد.

کلمات کلیدی

ماشین بردار پشتیبان، ماشین بردار پشتیبان دوقلو، مدل وزنی، وزن گرانشی، بیماری مزمن، دیابت، آلزایمر، سرطان

۱- مقدمه

در مطالعه‌ی داده‌های بیماری‌ها یکی از موضوعات با اهمیت، طبقه بندی آن‌هاست در رابطه با بیماری‌هایی مانند سرطان و دیابت، این طبقه‌بندی می‌تواند به صورت "سالم" و "ناسالم" بیان شود. با داشتن مجموعه‌ای از داده‌های ثبت شده، یکی ازروشهای پیش‌بینی طبقه‌ی نمونه‌ی جدید استفاده از ماشین‌های بردار پشتیبان (SVMs) است[1]. SVM یک تکنیک قدرتمند طبقه‌بندی است و سعی می‌کند داده‌ها را با استفاده از دو ابرصفحه‌ی موازی و ایجاد یک صفحه تصمیم‌گیری در بین آن‌ها از هم جدا کند. در بین نسخه‌های مختلف ارائه شده، ماشین‌های بردار پشتیبان دوقلو(Twin SVMs) که قید موازی بودن ابرصفحه ها را نادیده می‌گیرید[2] موفقیت بیشتری به دست آورده است. در کارهای نصیری و همکارن از لحاظ تئوری و هم به صورت کاربرد عملی در تشخیص حرکت انسان مورد استفاده قرار گرفته است[3],[4]. Xiong Si و Jing (2009) برای تشخیص توده در ماموگرافی دیجیتال استفاده کرده‌اند[5]. یکی از کاستی‌هایی که در svm استلندارد و twin svm وجود دارد این است که همه‌ی داده‌ها از اهمیت یکسان برخوردارند که در جهان واقعیت همیشه صادق نیست، برای غلبه بر این مشکل محققان بسیاری سعی کرده اند شیوه‌های مناسبی را ابداع کنند. برای طبقه بندی تومور Duan Hua و همکاران(2022) الگوریتمی با عنوان twin svm هیبریدی فازی را بکار برندند[6] که ترکیبی از مدل twin svm فازی است و با تولید یک ابرکره داده های مثبت و منفی را از هم جدا می‌کند. افزونه‌های بسیاری بر پایه‌ی اختصاص وزن به نمونه‌ها، برای این الگوریتم توسعه داده شده است که به الگوریتم‌های ماشین بردار پشتیبان دوقلوی وزنی(weighted Twin SVMs) مشهورند[7]. در این روش‌ها همچنان به نسبت بین کلاس‌ها اهمیت داده نمی‌شود در حالی که ممکن است داده هایی با نسبت‌های نامتوازن وجود داشته باشد مانند نسبت افراد بیمار به افراد سالم. اخیرا روشهایی برای کنترل نرخ عدم تعادل کلاس[2] ارائه شده است که با تعریف وزن به صورت نسبتی از این نرخ، سعی می‌کنند این مشکل را برطرف کنند اخیرا در یک مطالعه Xiaohan Yuan و دیگران(۲۰۲۲) نیز، یک چارچوب تشخیصی جدید برای بیماری‌های مزمن با داده‌های نامتوازن ارائه کردند[8]. از دیگر روش‌های ارائه شده در این زمینه، الگوریتم LSFLSTSVM-CIL است که در سال ۲۰۲۲ توسط M. A. Ganaie و همکاران توسعه داده شد[9]. این شیوه سعی می کند با استفاده از نرخ عدم تعادل کلاس، یک تابع وزن برای خطای هر نمونه از داده های کلاس بزرگتر تعریف کند در این روش و روش‌های دیگر، میزان اهمیت خطای کلاس کوچکتر برای همه نقاط داده یکسان در نظر گرفته می شود.

[1] Twin Support Vector machine

[2] Class Imbalance Rate

در این مقاله تابع وزن جدیدی بر پایه‌ی مفاهیم جاذبه در فیزیک بیان می‌شود، از این تابع در ضرایب خطای هر دو کلاس، برای به حداقل رساندن خطای تشخیص نادرست داده‌ها استفاده می‌شود، بنابراین اختصاص وزن به نمونه‌های کلاس کوچکتر باعث افزایش کارایی مدل می‌شود.

در ادامه، در بخش ۲ مفاهیم اساسی ماشین بردار پشتیبان دوقلو و وزنی معرفی می‌شود. در بخش ۳ روش مورد مطالعه و تابع وزن براساس مفهوم جاذبه‌ی گرانش بین دو جسم بیان می‌شود. در بخش ۴ به ارزیابی نتایج و مقایسه‌ی آن با روش‌های دیگر می‌پردازیم و از چندین مجموعه داده‌ی بیمای‌های مزمن برای آزمایش عملکرد آن استفاده می‌کنیم و نتایج بدست آمده را با چند روش دیگر مقایسه می شود. این مقایسه‌ها نشان می‌دهند که این روش می‌تواند با دقت بالاتری بیماری‌های مزمن را پیش‌بینی کند و انتظار می‌رود به روش‌های دیگر برتری داشته باشد.

۲- پژوهش‌های گذشته

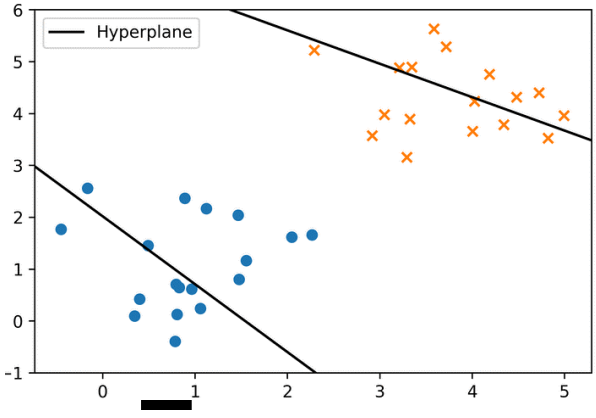
۲-۱- ماشین بردار پشتیبان دوقلو[1]

الگوریتم Twin SVM دو ابرصفحه غیرموازی را با حل دو مسئله‌ی برنامه ریزی درجه‌ی دو به صورت زیر بدست می آورد:

$$Min\ \frac{1}{2}\left(Aw^{(1)}+e_1b^{(1)}\right)^T\left(Aw^{(1)}+e_1b^{(1)}\right)+c_1e_2^Tq \quad (۱)$$
$$S.t.\ \ -\left(Bw^{(1)}+e_2b^{(1)}\right)+q\geq e_2,\ \ q\geq 0$$

$$Min\ \frac{1}{2}\left(Bw^{(2)}+e_2b^{(2)}\right)^T\left(Bw^{(2)}+e_2b^{(2)}\right)+c_2e_1^Tq \quad (۲)$$
$$S.t.\ \ \left(Aw^{(2)}+e_1b^{(2)}\right)+q\geq e_1,\ \ q\geq 0$$

که در روابط (۱) و (۲) A و B به ترتیب نمونه های کلاس مثبت و منفی هستند $b^{(i)}$ مقادیر اسکالر و $w^{(i)}$ بردارهای پارامترهای ابرصفحه ها، $c_i>0$ پارامترهای مدل، e_i بردارهایی با درایه های ۱ و q مقدار خطای ابرصفحه است. جمله‌ی اول در تابع هدف مجموع مربعات فاصله‌ی نقاط از ابرصفحه‌ی هر کلاس را پیاده سازی می کند و جمله ی دوم مجموع خطای متغیرها را کمینه می‌کند. و نمونه های جدید را براساس نزدیک بودن به ابرصفحه‌ها طبقه بندی می‌کند.



شکل ۱: ابرصفحه‌ی Twin SVM

۲-۲- ماشین بردار پشتیبان دوقلوی وزنی[2]

این مدل با تعریف وزن، اهمیت داده ها را در تعیین ابرصفحه ها در نظر می گیرد، با الهام از نظریه گراف ، یک گراف K نزدیکترین همسایه G را برای مدل سازی ساختار هندسی محلی داده‌ها ایجاد می کند. ماتریس وزن G را به صورت زیر تعریف می کند:

$$W_{ij}=\begin{cases}1 & x_i \text{ در } k-\text{همسایگی } x_j \text{ یا } x_j \text{ در } k-\text{همسایگی } x_i \text{ اگر}\\ 0 & \text{در غیر این صورت}\end{cases} \quad (۳)$$

براساس تعریف فوق برای مدل سازی فشردگی درون کلاسی و تفکیک پذیری بین طبقاتی دو ماتریس گراف برای هریک از جفت TWSVM می سازد، یک گراف درون کلاسی G_s و یک گراف بین کلاسی G_d. این دو زیرگرافهایی از G هستند.

$$W_{s,ij}=\begin{cases}1 & \text{اگر حداقل یکی از } x_i \text{ و } x_j \text{ در همسایگی دیگری در } G_s \text{ باشد}\\ 0 & \text{در غیر این صورت}\end{cases} \quad (۴)$$

$$W_{d,ij}=\begin{cases}1 & \text{اگر حداقل یکی از } x_i \text{ و } x_j \text{ در همسایگی دیگری در } G_d \text{ باشد}\\ 0 & \text{در غیر این صورت}\end{cases} \quad (۵)$$

ایده WLTSVM کشف اطلاعات شباهت ذاتی در نمونه‌های یک کلاس و استخراج بردارهای پشتیبان احتمالی موجود در نمونه‌های کلاس دیگر است. ماتریس وزن را با بازتعریف از روی روابط بالا به صورت زیر می سازد:

$$f_j=\begin{cases}1 & \text{اگر وجود داشته باشد } i \text{ که } W_{d,ij}\neq 0\\ 0 & \text{در غیر این صورت}\end{cases} \quad (۶)$$

مشابه TWSVM سعی می کند دو صفحه غیر موازی بدست آورد که هر کدام با نقاط کلاس مربوطه مطابقت دارند. به دنبال تفسیر هندسی مشابه، یک مسئله بهینه سازی برای تخمین ابرصفحه کلاس +۱ و −۱به صورت زیر بیان می شود:

$$\text{Min}\ \frac{1}{2}\sum_{i=1}^{N_1}\sum_{j=1}^{N_1}W_{s,ij}(w_1^Tx_j^{(1)}+b_1)^2+C\sum_{j=1}^{N_2}\xi_j, \quad (۷)$$
$$\text{s.t.}\ -f_j(w_1^Tx_j^{(2)}+b_1)+\xi_j\geq f_j\cdot 1,\quad \xi_j\geq 0.$$

$$\text{Min}\ \frac{1}{2}\sum_{j=1}^{N_1}d_j(w_1^Tx_j^{(1)}+b_1)^2+C\sum_{j=1}^{N_2}\xi_j, \quad (۸)$$
$$\text{s.t.}\ -f_j(w_1^Tx_j^{(2)}+b_1)+\xi_j\geq f_j\cdot 1,\quad \xi_j\geq 0$$

که در روابط (۷) و (۸) عبارت $f_j.\mathbf{1}$ به معنی ضرب اسکالر $\boldsymbol{f_j}$ در برداری با درایه‌های ۱ است.

۳- روش پیشنهادی

الگوریتم LSFLSTSVM-CIL[3] مانند مدل پایه‌ی Twin SVM و WLTSVM دو ابرصفحه‌ی غیرموازی را برای جداسازی کلاس‌ها از یکدیگر جستجو می‌کند برای کاهش خطای کلاس بزرگتر نیز در تابع هدف، از یک تابع وزن [10] برای کاهش مجموع مربعات خطای آن کلاس استفاده می کند. در اینجا به بیان خلاصه‌ی از این الگوریتم می‌پردازیم. خواننده می‌توانند برای جزئیات بیشتر به [9] مراجعه کند.

[1] Twin Support Vector Machine (TSVM or Twin SVM)
[2] Weighted Twin Support Vector Machine (WTSVM)
[3] Large Scale Fuzzy Least Squares Twin SVM-Class Imbalance Learning

۳-۱- حالت خطی الگوریتم LSFLSTSVM-CIL

همانطور که گفته شـــد این الگوریتم دو ابرصـــفحه‌ی غیرموازی را برای جداسازی داده‌ها جستجو می‌کند که به صورت زیر داده می شود:

$$\begin{cases} w_1^T x + b_1 = 0 \\ w_2^T x + b_2 = 0 \end{cases} \quad (۹)$$

که در رابطه‌ی فوق $w_1, w_2 \in \mathbb{R}^n$ و $b_1, b_2 \in \mathbb{R}$ و x یک نمونه‌ی دلخواه از داده هاست، تابع هدف مسئله به صورت زیر تعریف می شود:

$$\min_{w_1,b_1,\xi_2,\eta_1} \frac{c_3}{2}(||w_1||^2 + b_1^2) + \frac{1}{2}\eta_1^T\eta_1 + \frac{c_1}{2}(S_2\xi_2)^T(S_2\xi_2)$$

$$\boldsymbol{S.t.}\ \ Aw_1 + eb_1 = \eta_1 \\ -(Bw_1 + eb_1) + \xi_2 = e \quad (۱۰)$$

$$\min_{w_2,b_2,\xi_1,\eta_2} \frac{c_4}{2}(||w_2||^2 + b_2^2) + \frac{1}{2}\eta_2^T\eta_2 + \frac{c_2}{2}(S_1\xi_1)^T(S_1\xi_1) \quad (۱۱)$$

$$\boldsymbol{S.t.}\ \ Bw_2 + eb_2 = \eta_2 \\ (Aw_2 + eb_2) + \xi_1 = e$$

که A و B به ترتیب نمونه‌های کلاس کوچکتر و بزرگتر، w_i, b_i پارامترهای ابرصـــفحه ، ξ_i متغیرهای کمکی e بردار متشـــکل از درایه‌های برابر ۱ ، c_i ابرپارامترهای مدل ، S_2 ماتریس قطری شـــامل وزن خطاهای کلاس بزرگتر و S_1 ماتریس همانی است.

مسئله های بهینه سازی فوق با استفاده از ضرایب لاگرانژ و اعمال شرایط K.K.T به صورت روابط (۱۲) و (۱۳) در می آید[9].

$$\max_{\alpha,\beta} -\frac{1}{2}(\alpha^t \ \ \beta^t)\,\hat{Q}\,(\alpha^t \ \ \beta^t)^t - c_3\beta^t e \quad (۱۲)$$

$$\hat{Q} = \begin{bmatrix} AA^t + c_3 I & AB^t \\ BA^t & BB^t + \frac{c_3}{c_1}(S_2^{-1})^2 \end{bmatrix} + E$$

$$\max_{\lambda,\theta} -\frac{1}{2}(\lambda^t \ \ \theta^t)\,Q'\,(\lambda^t \ \ \theta^t)^t - c_4\theta^t e \quad (۱۳)$$

$$Q' = \begin{bmatrix} BB^t + c_4 I & BA^t \\ AB^t & AA^t + \frac{c_4}{c_2}(S_1^{-1})^2 \end{bmatrix} + E$$

و بردارهای w_i و مقادیر b_i از رابطه‌ی (۱۴) بدست می‌آید که در این رابطه α و β مقادیر ضرایب لاگرانژ هستند.

$$\begin{bmatrix} w_1 \\ b_1 \end{bmatrix} = \frac{1}{c_3}\begin{bmatrix} A^t & B^t \\ e^t & e^t \end{bmatrix}\begin{bmatrix} \alpha \\ \beta \end{bmatrix} \quad (۱۴)$$

پس از حل مسئله‌ی فوق تابع تصمیم گیری برای پیش‌بینی طبقه‌ی نمونه‌ی جدید از رابطه‌ی زیر استفاده می شود:

$$class(x) = \underset{i=1,2}{argmin}(|x^T w_i + b_i|) \quad (۱۵)$$

۳-۲- حالت غیرخطی الگوریتم LSFLSTSVM-CIL

استفاده از توابع هسته[۱] در داده‌هایی که به صورت خطی قابل تفکیک نیستند یکی از ترفندهای مفید اســـت[12]. برای حالت غیر خطی با انتخاب یک هســـته‌ی مناســـب، ماتریس A با $K(A, A^T)$ و ماتریس B با $K(B, B^T)$ ماتریس AB^T با $K(A, B^T)$ و ماتریس BA^T با $K(B, A^T)$ جایگزین می شود[9]. با جایگزینی موارد ذکر شده تابع تصمیم به صورت زیر اصلاح می شود:

$$f_1(x) = \frac{1}{c_3}\left[K(x, A^t),\ K(x, B^t)\right]\begin{bmatrix}\alpha \\ \beta\end{bmatrix} + b_1 \quad (۱۶)$$

$$b_1 = \tfrac{1}{c_3}(e^t\alpha + e^t\beta)$$

$$f_2(x) = \frac{1}{c_4}\left[K(x, B^t),\ K(x, A^t)\right]\begin{bmatrix}\lambda \\ \theta\end{bmatrix} + b_2 \quad (۱۷)$$

$$b_2 = -\tfrac{1}{c_4}(e^t\lambda + e^t\theta)$$

$$class(x) = \underset{i=1,2}{argmin}(|f_i(x)|) \quad (۱۸)$$

و کلاس نمونه‌ی جدید مطابق رابطه‌ی (۱۸) پیش بینی می‌شود.

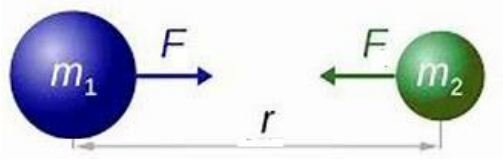
۳-۳- تابع وزن

در مدل‌های پیشـین چون مقادیر وزن به صـورت ضـرایب خطای نمونه های کلاس بزرگتر درنظر گرفته شـــده لند فقط منجر به کاهش خطای نمونه‌های کلاس بزرگتر می‌شـود و برای داده‌های کلاس کوچکتر وزنی اختصـاص داده نمی‌شود، در داده‌های بیماری‌های مزمن مثل سـرطان و دیابت و بیماری‌های ویروســـی همه‌گیر مانند COVID-19 اهمیت بســـیار زیادی وجود دارد که خطای خیلی کمتری در تشـــخیص کلاس کوچکتر اتفاق بیافتد در این داده‌ها تشخیص اشتباه یک نمونه به معنای این است که یک شخص ناسالم را سالم تشخیص دهیم. و این هزینه‌ی گزافی خواهد داشت. ما در اینجا ابتدا به اصلاح تابع وزن استفاده شده در مدل فوق اقدام کردیم و سپس با تخصیص وزن به داده‌های کلاس کوچکتر خطای این داده‌ها را کاهش دادیم.

۳-۴- تعریف تابع وزن

در فیزیک نیروی گرانش F بین دوجسم به جرم m_1 و m_2 با فاصله‌ی r از یکدیگر و با ثابت گرانش G به صورت زیر تعریف می شود:

$$F = G\frac{m_1 \times m_2}{r^2} \quad (۱۹)$$



شکل ۲- برهم کنش دو جرم در فیزیک

[۱] Kernel Functions

با بهره گیری از این مفهوم و اندکی تغییر در آن، برای نمونه‌ی x وزن خطای آن را به صورت زیر تعریف می کنیم:

$$weight(x) = G\frac{N_1 \times N_2}{r(x)+1} \qquad (۲۰)$$

که در رابطه‌ی فوق N_1 تعداد نمونه‌ها در همسایگی نمونه‌ی x به همراه خود آن، و N_2 نیز همین مفهوم برای مرکز کلاس به شعاع ε است ، در رابطه‌ی فوق $r(x)$ فاصله‌ی نمونه‌ی x از مرکز کلاس است. این رابطه برای محاسبه‌ی عناصر قطری ماتریس های S_1 و S_2 استفاده می‌شود، این درحالی است که در مرجع [9]، [10] و [12] فقط به ماتریس S_2 وزن داده می‌شود و S_1 یک ماتریس همانی در نظر گرفته شده است. مقدار IR برای نقاط کلاس کوچکتر برابر یک و برای نقاط کلاس بزرگتر به صورت رابطه‌ی زیر تعریف می‌شود:

$$IR = \frac{number\ of\ samples\ in\ class\ B}{number\ of\ samples\ in\ class\ A} \qquad (۲۱)$$

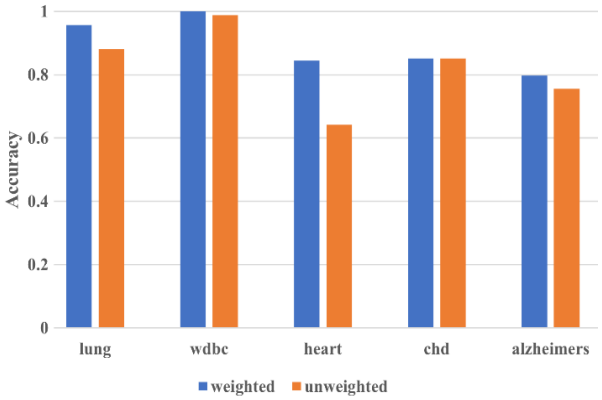
در اینجا نمونه‌ی x مانند یک جسم با جرم N_1 در نظر گرفته می‌شود که عضویتش در کلاس تحت تاثیر مرکز کلاس با جرم N_2 می باشد. و از این رو متناسب با وزنش در تعیین ابرصفحه نقش ایفا می‌کند. در رابطه‌ی (۲۰) اهمیت نمونه‌ی x با توجه به تعداد نقاط اطراف و همچنین فاصله‌ی آن از مرکز کلاس سنجیده می‌شود. هر چه تعداد نقاط اطراف یک نمونه بیشتر و به مرکز کلاس نزدیک‌تر باشد دارای وزن بیشتری در این کلاس خواهد بود، می‌دانیم فاصله‌ی یک داده‌ی پرت از مرکز کلاس زیاد است علاوه بر این به دلیل دورافتادگی، در اطراف خود نقاط بسیار اندکی دارد و یا یک نقطه‌ی کاملا تنهاست. با این توصیفات چنین نقطه‌ای وزن بسیار اندکی خواهد داشت و بنابراین از اهمیت کمتری برخوردار است.

جدول ۱: نسبت عدم تعادل کلاس

Dataset	Lung	wdbc	Heart	chd	Alzheimer
Imbalance ratio	7.0	1.6	1.23	5.52	2.33

۴- نتایج داده‌های آزمایشی

در این بخش عملکرد روش خود را روی چند نمونه از داده‌های بیمای‌های مزمن مورد ارزیابی قرار می‌دهیم، برای حالت غیرخطی از کرنل گوسی $K(x_1,x_2) = e^{-\mu||x_1-x_2||^2}$ و برای حالت خطی از کرنل خطی $K(x_1,x_2) = x_1.x_2^T$ استفاده کردیم. مقادیر پارامترها ازجدول ۱ و با استفاده از جستجوی تصادفی بدست آمده است، برای ارزیابی مدل از روابط (۲۲) تا (۲۶) استفاده شده است. در شکل ۳ عملکرد مدل در دو حالت مورد بررسی قرار گرفته است. یک بار به داده‌های هر دو کلاس وزن تخصیص داده شده و در حالت دوم وزن داده‌های کلاس کوچکتر برابر یک در نظر گرفته شده است. نسبت عدم تعادل این داده‌ها در جدول ۲ نشان داده شده است. در



شکل ۳: تفاوت دقت در دو حالت وزن دهی

شکل ۳ دیده می‌شود مدل مورد مطالعه در ۴ مورد از ۵ مورد نتایج بهتر و در یک مورد دارای نتیجه‌ی برابر بدست آورده است. می توان انتظار داشت وقتی به کلاس کوچکتر وزن داده شود نتایج بهتری کسب شود.

جدول ۲: پارامترهای مدل

پارامترهای مدل	محدوده‌ی مقادیر
c, c_0	$0.25 \times i\ ,\quad i: 1,\dots,10$
c_1, c_2, c_3, c_4, μ	$10^i, i: -5, \dots 5$
r	$\begin{cases} \frac{10^{-5}}{10^i} & i: -4,-3,-2,-1,0 \\ 0.1+\frac{i}{10} & i: 1,\dots,9 \end{cases}$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \qquad (۲۲)$$

$$Recall = \frac{TP}{TP+FN} \qquad (۲۳)$$

$$Precision = \frac{TP}{TP+FP} \qquad (۲۴)$$

$$F-score = \frac{2 \times Precision \times Recall}{Precisoin + Recall} \qquad (۲۵)$$

$$G-mean = \sqrt{Precision \times Recall} \qquad (۲۶)$$

جدول ۳ مقایسه‌ی بین نتایج این مدل با نتایج چند نسخه از توسعه‌های الگوریتم twin svm که به طور ویژه برای داده‌های نامتوازن طراحی شده اند را روی داده‌های سرطان و دیابت نشان می دهد. این داده‌ها در مخزن داده های keel[۱] و Kaggle[۲] در دسترس است. نتایج نشان می‌دهد در داده‌های سرطان، مدل مورد مطالعه به دقت ۱۰۰٪ دست یافته و در داده‌های دیابت رتبه‌ی دوم را دارد. در رابطه با یک بیماری همه‌گیر مانند COVID-19 و یا انواع سرطان، اگر شخص سالمی بیمار تشخیص داده شود می توان محتاطانه تدابیر لازم را اتخاذ کرد، اما اگر شخص بیماری سالم تشخیص داده شود ممکن است عواقب جبران ناپذیری داشته باشد در این

[۱] https://sci2s.ugr.es/keel/development.php

[۲] https://kaggle.com

جدول ۳: مقایسه‌ی دقت مدل مورد مطالعه در نمونه داده‌های Pima و Breast Cancer با ٤ مدل دیگر

Datasets	HFSWLSTSVM*	KWRUTSVM-CIL*	RFLSTSVM-CIL* (c_0, c_1, μ)	IFW-LSTSVM-CIL* (c_1, c_2, k, μ)	**This study** $(c, c_1, c_2, c_3, c_4, \mu, r)$
Breast Cancer	98.55	98.01	98.76 (1.5,0.1,16)	99.07 $(10^{-2}, 10^{-4}, 1, 16)$	**100** $(0.75, 2.25, 10, 10^{2}, 10^{-4}, 10^{-5}, 10^{-5}, 10^{-5})$
Pima Indian	**89.71**	71	62.77 (0.5,0.1,32)	76.77 (0.001,0.1,1,2)	79.22 $(1.25, 0.5, 10^{2}, 10^{2}, 10^{-4}, 10^{-3}, 0.1, 0.1)$

* این داده‌ها از [13]، [14] و [15] بدست آمده است.

مورد اهمیت معیار recall بسیار زیاد است، این معیار با توجه به رابطه‌ی (۱۵) به معنی دقت مدل در تشخیص درست شخص واقعا بیمار است.

جدول ۴ عملکرد مدل را با این معیار مورد ارزیابی قرار داده است. در این جدول مشاهده می شود مدل علاوه بر نتایج مطلوب معیار صحت(Accuracy)، هماهنگ با آن نتایج معیار Recall نیز بسیار مطلوب است. در مورد Lung با میزان دقت ٪۹۵.۷۰ معیار Recall نشان دهنده‌ی این است که مدل ٪۹۱.۶۷ از بیماران را به درستی تشخیص داده است. در رابطه با داده‌های heart و hepatitis مقدار recall حتی بهتر از accuracy است و در سایر موارد نیز بیشترین تفاوت بین این دو معیار در مورد آلزایمر تقریبا ٪۹ است. این مقایسه نشان می دهد روش مورد مطالعه در هر سه مورد نتایج بهتری نشان می دهد. از نظر مقدار recall در دیتاست Pima ضعیف بوده و در hepatitis با اختلاف تقریبا ٪۵ برتری دارد.

کوچکتر در بیماریهای مزمن بسیار بیشتر از گروه بزرگتر است توجه ویژه داشتیم تا بتوانیم با افزای دقت تشخیص صحیح در این بخش اعتبار مدل را بهبود ببخشیم، نتایج بررسی شده نشان داد که اعمال وزن به خطای گروه کوچکتر می تواند اهمیت این گروه را در تعیین طبقه بند صحیح منعکس کند با توجه به مطالب بررسی شده و نتایج موجود و مقایسه‌های انجام گرفته می‌توان انتظار داشت ماشین بردار پشتیبان دوقلوی با وزن گرانشی(روش مورد مطالعه) با اختصاص وزن به هر دو کلاس، در داده‌های نامتوازن عملکرد بهتری دارد. از طرفی با در نظر گرفتن مقدار نسبت عدم تعادل در گروه بزرگتر کنترل سوگیری مدل را مانند مدلهای ذکر شده حفظ می‌کند به ویژه در داده‌های بیماری‌ها که اغلب اهمیت کلاس کوچکتر از کلاس بزرگتر بیشتر است این مدل عملکرد بهتری نسبت به سایر مدل‌ها دارد.

۵- نتیجه گیری

در این مقاله در رابطه با داده‌های نامتوازن و اهمیت آن‌ها در الگوریتم‌های طبقه‌بندی برپایه‌ی twin svm بحث کردیم. به این نکته که اهمیت گروه

جدول ٤: عملکرد مدل و معیارهای ارزیابی

Dataset $(c, c_1, c_2, c_3, c_4, \mu, r)$	IR	Kernel	Accuracy	Recall	Precision	f-score	g-mean
Lung $(1,1.25,1,10^{3},10^{-5}, 10^{-3}, 10^{-4}, 10^{-5})$	7	rbf	**0.9569**	**0.9167**	0.7857	0.8461	0.8487
Breast Cancer $(0.75,2.25,10,10^{2}, 10^{-4}, 10^{-5}, 10^{-5}, 10^{-5})$	1.63	rbf	**1**	**1**	1	1	1
Pima $(1.25,0.5, 10^{2}, 10^{2}, 10^{-4}, 10^{-3}, 1,1)$	1.98	linear	0.7705	0.7159	0.6923	0.7039	0.7040
Ckd $(1,1.75, 10^{4}, 10^{5}, 10^{3}, 10^{-2}, 10^{-2}, 0.2)$	1.74	linear	**1**	**1**	1	1	1
Hepatitis $(0.5,1.25, 10^{2}, 10^{2}, 10^{5}, 10, 10^{-5}, 10^{-1}, 10^{-4})$	3.32	linear	**0.9149**	**1**	0.6364	0.7778	0.7977
Heart $(0.25,2,1,10^{2}, 10^{2}, 10^{-4}, 10^{-3}, 0.3)$	1.23	linear	**0.8406**	**0.8689**	0.7910	0.8290	0.7516
Alzheimer $(1.75,2, 10^{-2}, 10^{-1}, 1,0.1,0.1,0.1)$	2.33	rbf	0.7972	0.7025	0.6967	0.6996	0.6996

مراجع

[1] V. Vapnik ",*Support Vector Networks*", Nature of Statistical Learning Theory (ser. Statistics for Engineering and Information Science). New York, NY, USA: Springer, 2000.

[2] JA Nasiri, AM Mir," *An Enhanced KNN-based twin support vector machine with stable learning rules*", Neural computing and applications 32 (16), 12949-12969,2020.

[3] K Mozafari, JA Nasiri, NM Charkari, S Jalili, "*Informatics and Computational Intelligence (ICI), 2011.*

[4] Jayadeva, R. Khemchandani, and S. Chandra, "*Twin support vector machines for pattern classification*", IEEE Trans. Pattern Anal. Mach. Intell., vol. 29, no. 5, pp. 905–910, May 2007.

[5] Xiong Si, Lu Jing, "*Mass Detection in Digital Mammograms Using Twin Support Vector Machine-based CAD System*", WASE International Conference on Information Engineering, 2009.

[6] DUAN Hua1, FENG Tong1, LIU Songning1, ZHANG Yulin1 , and SU Jionglong, "*Tumor Classification of Gene Expression Data by Fuzzy Hybrid Twin SVM*", Chinese Journal of Electronics Vol.31, No.1, Jan. 2022.

[7] Qiaolin Yea, Chunxia Zhaoa, Shangbing Gaoa, Hao Zheng," *Weighted Twin Support Vector Machines with Local Information and its application*", Neural Networks, 31-39, 2012.

[8] XiaohanYuan, Shuyu Chen, Chuan Sun & LuYuwen, "*A novel early diagnostic framework for chronic diseases with class imbalance*", Scientifc Reports, 12:8614, 2022.

[9] M. A. Ganaie, M. Tanveer, Senior Member, IEEE, and Chin-Teng Lin, *"Large-Scale Fuzzy Least Squares Twin SVMs for Class Imbalance Learning",*IEEE TRANSACTIONS ON FUZZY SYSTEMS, VOL. 30, NO. 11, NOVEMBER 2022.

[10] B. Richhariya and M. Tanveer, "*A robust fuzzy least squares twin support vector machine for class imbalance learning*", Appl. Soft Comput., vol. 71 ,pp. 418–432, 2018.

[11] Mokhtar S. Bazaraa John J. Jarvis Hanif D ",*Linear Programming and Network Flows",* New York: Wiley,1977.

[12] Avrim Blum, John Hopcroft, and Ravindran Kannan, "*Foundations of Data Science*", Thursday 4th January, 2018.

[13] Divya Tomar and Sonali Agarwal, "*Hybrid Feature Selection Based Weighted Least Squares Twin Support Vector Machine Approach for Diagnosing Breast Cancer, Hepatitis, and Diabetes*", Hindawi Publishing Corporation Advances in Artificial Neural Systems Volume 2015.

[14] M.A. Ganaie, M. Tanveer, "*KNN weighted reduced universum twin SVM for class imbalance learning*",

[15] M. Tanveer, Senior Member, IEEE, M. A. Ganaie, A. Bhattacharjee, and C. T. Lin, "*Intuitionistic Fuzzy Weighted Least Squares Twin SVMs*", IEEE.

پنجمین کنفرانس ملی انفورماتیک ایران
پژوهشگاه دانشهای بنیادی، پردیس فرمانیه، تهران
۱۳ و ۱۴ دی ماه ۱۴۰۲

بهبود عملکرد یافتن نوع مشتری با رویکرد چندمرحله‌ای در صنعت هتلداری

حامد شرافت مولا[۱]، هادی یعقوبیان[۲]، راضیه ملک حسینی[۳]، کرم الله باقری فرد[۴]

[۱] دانشکده فنی مهندسی، گروه کامپیوتر، واحد یاسوج، دانشگاه آزاد اسلامی، یاسوج
h.sherafat.m@gmail.com

[۲] دانشکده فنی مهندسی، گروه کامپیوتر، واحد یاسوج، دانشگاه آزاد اسلامی، یاسوج – نویسنده مسئول
باشگاه پژوهشگران جوان و نخبگان، دانشگاه آزاد اسلامی، واحد یاسوج، یاسوج
yaghoobian.h@gmail.com

[۳] دانشکده فنی مهندسی، گروه کامپیوتر، واحد یاسوج، دانشگاه آزاد اسلامی، یاسوج
باشگاه پژوهشگران جوان و نخبگان، دانشگاه آزاد اسلامی، واحد یاسوج، یاسوج
malekhoseini.r@gmail.com

[۴] دانشکده فنی مهندسی، گروه کامپیوتر، واحد یاسوج، دانشگاه آزاد اسلامی، یاسوج
باشگاه پژوهشگران جوان و نخبگان، دانشگاه آزاد اسلامی، واحد یاسوج، یاسوج
ka.bagherifard@iau.ac.ir

چکیده

امروزه سیستم‌های «مدیریت درآمد» به یک ضرورت تبدیل شده‌اند و این سیستم‌ها بر دقت «برآورد تقاضا» متکی هستند. یکی از جنبه‌های کلیدی برآورد تقاضا، شناسایی انواع مشتریان است که با عنوان مسئلهٔ «کشف نوع مشتری» شناخته می‌شود. این امر اخیراً توسط الگوریتم‌های فراابتکاری ژنتیک و ممتیک حل‌شده است. معمولاً روش‌های فراابتکاری برای حل مسئله به جمعیت اولیه نیاز دارند و شروع کار الگوریتم با جمعیت اولیه ای بهینه می‌تواند عملکرد الگوریتم را در یافتن پاسخ بهبود بخشد. البته باید به این نکته توجه کرد که ایجاد جمعیت اولیه ای بهینه می‌تواند برای بعضی از مسئله‌ها چالش‌برانگیز باشد. در این تحقیق، رویکردی چندمرحله ای جدید ارائه می‌دهیم که فهرستی از «محصولات مؤثر» را ایجاد می‌کند و با استفاده از آن مجموعه‌ای از انواع مشتری را برای جمعیت اولیه ایجاد می‌کنیم. ما رویکرد خود را برای ارزیابی روی پنج مجموعه‌داده واقعی هتل‌های زنجیره‌ای کانتینتال ایالات‌متحده آزمایش می‌کنیم و نشان می‌دهیم که رویکردمان نسبت به جمعیت اولیه تصادفی در همگرایی به صورت میانگین ۴۳٪ بهتر عمل می‌کند.

کلمات کلیدی

مدیریت درآمد، الگوریتم‌های فراابتکاری، الگوریتم ژنتیک، الگوریتم ممتیک، جمعیت اولیه

۱- مقدمه

ایجاد بیشترین سود یا درآمد را می‌توان از طریق سیستم‌های مدیریت سود به دست آورد. مفهوم مدیریت سود پس از مقررات‌زدایی از صنعت هوانوردی ایالات متحده در سال ۱۹۷۸ معرفی شد [۱]. سیستم‌های مدیریت سود به شرکت‌ها در تصمیم‌های مهمی که بر رفتار خرید مشتری تأثیر می‌گذارد، کمک می‌کنند. این تصمیم‌ها را می‌توان به سه دسته تقسیم کرد: برآورد تقاضا، تعیین محصول برای فروش و استراتژی قیمت‌گذاری. مهم‌ترین موضوع در فرایند مدیریت درآمد، برآورد تقاضاست؛ زیرا در برآورد تقاضا مشخص می‌شود که‌چه نوع مشتری وارد بازار می‌شود و کدام محصول را خریداری خواهد کرد [۲]–[۵]. برای بدست آوردن برآورد تقاضا، باید درک روشنی از ترجیحات و علایق مشتری داشته باشیم که آن‌ها را می‌توانیم از دادهٔ فروش قبلی بدست آوریم. برای پیداکردن نوع‌های مشتری، آن‌ها را با لیستی شخصی‌سازی‌شده از محصولاتی که ترجیح می‌دهد بخرد، نشان می‌دهیم [۸] [۱۳]–[۱۵]. مشکل اصلی اینجاست که اگر n محصول داشته باشیم، تعداد بالقوه انواع مشتری با افزایش تعداد محصولات و امکان جایگزینی محصولات برای مشتری به‌صورت فاکتوریل افزایش می یابد [۱۶], [۱۷]. برای حل این مشکل، باید انواع مشتریان را از تراکنش‌های فروش و داده‌هایی که مشخص می‌کند چه محصولاتی برای خرید در دسترس بوده‌اند، مشخص کرد. پیداکردن انواع مشتری که با عنوان مسئلهٔ «کشف نوع مشتری» نیز شناخته می‌شود، باتوجه‌به تعداد محصولات و تمامی انواع مشتری که ممکن است وجود داشته باشد، مسئله‌ای از نوع NP-Hard است [۱۶]. روش‌های ریاضی مختلفی برای حل مشکل شناسایی انواع مشتری استفاده‌شده است، بیشتر این روش‌ها

از مدل‌های «برآورد تقاضا مبتنی بر انتخاب» استفاده می‌کنند [۱۶]–[۱۸]. اخیرا با استفاده از الگوریتم‌های فراابتکاری این مسئله حل‌شده است. در [۱۹] از الگوریتم ژنتیک و در ادامه در [۳۴] از الگوریتم ممتیک برای حل مسئلهٔ «کشف انواع مشتری» استفاده‌شده است. باید به این نکته دقت کرد که هر دو رویکرد ذکرشده می‌توانند زمان‌بر باشند و امکان استفاده از آن‌ها در برنامه‌های تجاری که نیاز سریع به نتیجه دارند، به همین دلیل ممکن نباشد. ساخت جمعیت اولیه مرحله‌ای رایج در همه الگوریتم‌های فراابتکاری است [۲۶] [۲۷]. هدف این مرحله ارائهٔ برآوردی اولیه از راه‌حل هاست و الگوریتم این راه‌حل ها را به صورت مداوم بهبود می بخشد تا زمانی که شرایط توقف برآورده شود. جمعیت اولیه خوب می تواند به الگوریتم کمک کند تا به جواب بهینه برسد؛ در حالی که جمعیت اولیهٔ نامناسب می تولند الگوریتم را در یافتن جواب بهینه ناموفق سازد [۲۶] [۲۷] [۳۳]. تحقیقات نشان داده است که نتایج الگوریتم‌های فراابتکاری به دو مورد بستگی دارد: ویژگی‌های الگوریتم و جمعیت اولیه [۲۸] [۲۹]. یکی از راه‌های بهبود جمعیت اولیه، استفاده از کروموزوم‌هایی باکیفیت بیشتر است [۳۰] [۳۱] و همچنین محققان افزایش تنوع جمعیت اولیه را برای کشف فضای جست وجوی بیشتر مناسب دانسته‌اند [۲۷]. جمعیت اولیه برای بهینه سازهای فراابتکاری معمولاً تصادفی و بر اساس اصل فضای جست وجو است [۲۷] [۳۱] [۳۲]. ازآنجایی‌که کیفیت جمعیت اولیه در ابتدای الگوریتم پیش بینی ناپذیر است، پژوهشگران روش های مختلف ساخت جمعیت اولیه را برای پوشش فضا و حداکثرسازی فضای یک مسئلهٔ معین و بررسی می‌کنند. هدف این مطالعه پیشنهاد رویکردی برای ساخت جمعیت اولیه برای مسئله کشف نوع مشتری است. این جمعیت اولیه باعث همگرایی سریع‌تر از راه‌حل‌های غیرقابل‌قبول اولیه به راه‌حل‌های قابل‌قبول در الگوریتم‌های ژنتیک و ممتیک برای حل مسئله کشف نوع مشتری می‌شود و اولین راه‌حل قابل‌قبول نسبت به جمعیت اولیه تصادفی سریع‌تر پیدا می‌شود و موجب بهبود فرایند یافتن راه‌حل بهینه خواهد شد.

۲- بررسی ادبیات

ون رایزین و ولکانو مدل تخمین تقاضای ناپارامتریک خود را در مقالهٔ [۱۶] برای شناسایی انواع مشتری با استفاده از تراکنش‌های فروش و داده‌های دردسترس‌بودن محصول پیشنهاد کردند. این مدل شامل سه مرحله است: در مرحلهٔ اول، یک لیست محصول تکی را برای نشان‌دادن لیست ترجیحی مشتری که شامل یک محصول و صفر (عدم خرید) استفاده می‌شود را ایجاد کردند. در مرحله دوم، از روش «حداکثرسازی راست‌نمایی تخمین[۱]» برای ارزیابی برازندگی هر راه‌حل با استفاده از مدل برآورد تقاضا استفاده می‌شود. در انتها، برنامهٔ عدد صحیح مختلط[۲] برای شناسایی یک نوع مشتری جدید استفاده می‌شود که با ادغام در راه‌حل فعلی، برازندگی آن را افزایش می‌دهد. این روند تا زمانی تکرار می‌شود که مدل دیگر نتواند نوع مشتری جدیدی را پیدا کند که برازندگی را بهبود می بخشد. آن ها در کار بعدی خود [۱۷] باهدف ساده‌سازی و افزایش کارایی مدل خود، از روش بیشینه‌سازی انتظارات[۳] برای حل مشکل حداکثرسازی راست‌نمایی تخمین استفاده کردند. در [۲۰]، مدل برآورد تقاضای مبتنی بر انتخاب[۴] برای پیداکردن انواع مشتری از طریق سه مرحله اعمال می‌شود: ساخت یک نمودار غیر چرخه‌ای جهت‌دار برای هر نوع مشتری، گروه‌بندی انواع مشتریان مشابه و تخمین توزیع ترجیحات مشتری. نویسندگان از داده‌های تابلویی[۵] استفاده کردند که گسترده‌تر و دقیق‌تر از داده‌های تراکنش‌های فروش و دردسترس‌بودن محصول است. نویسندگان [۲۲] یک رویکرد سه‌بخشی برای کشف انواع مشتری بر اساس مسئلهٔ سفارش خطی کلاسیک[۶] پیشنهاد کردند. قسمت ۱ یک راه‌حل اولیه ایجاد می‌کند، قسمت دوم PMF هر نوع مشتری را تخمین می‌زند و قسمت ۳ نوع مشتری جدید را شناسایی می‌کند. این سه قسمت تا زمانی که معیارهای توقف برآورده شود تکرار می‌شود. نویسندگان در [۳۴] رویکردی برای تعیین تعداد تکرار هر راه‌حل قابل‌قبول در فضای جست وجو معرفی کردند. این رویکرد می تواند به جست وجوی محلی مؤثرتر در الگوریتم تکاملی شود. آنها الگوریتم ممتیک ارائه کردند که از داده‌های فراوانی برای کشف انواع مشتری استفاده می‌کرد.

۳- توضیح مسئله کشف نوع مشتری

مسئله‌ای که دربارهٔ آن صحبت خواهیم کرد، کشف نوع مشتری است که هدفش شناسایی انواع مشتریان ناشناخته بر اساس محصولاتی است که در دوره‌های زمانی متعدد خریده شده است. در نظر بگیرید n محصول که به‌صورت $N=\{0, 1, 2,... , n\}$ نمایش داده می‌شود، توسط انواع مشتریان ناشناخته در دوره‌های زمانی T خریداری‌شده است ($t = \{1, 2,... , T\}$). اگر مشتری محصول صفر را انتخاب کند، یعنی چیزی خریداری نکرده و بازار را ترک می‌کند. هر نوع مشتری با لیست ترجیحی محصولاتی تعریف می شود که قصد خرید دارد و محصول h اولویت بیشتری نسبت به j دارد ($C(h) < C(j)$)، به این معنی که مشتری خرید محصول h را به j ترجیح می‌دهد. لیست محصولات ارائه‌شده در هر دورهٔ زمانی با $St \subseteq N$ نشان داده می‌شود و St باید حداقل دو محصول (محصول صفر و یک محصول دیگر) داشته باشد. هنگامی که یک نوع مشتری در بازه زمانی وارد بازار می‌شود، باتوجه‌به لیست اولویت‌های خود، از لیست محصولات ارائه شده، محصولی را انتخاب می کند که بیشترین اولویت را در لیست ترجیحی دارد. اگر نتواند محصولی را بخرد، محصول صفر را انتخاب کرده و بازار را ترک می‌کند. نرخ ورود مشتریان در هر دوره یکنواخت با احتمال $0 < \lambda < 1$ است که در تمام دوره‌ها یکسان در نظر گرفته می‌شود. جدول ۱ نمونه‌ای از بازه‌های زمانی و اطلاعات موجود بودن محصول را نشان می‌دهد. این جدول شامل پنج بازهٔ زمانی و پنج محصول است. در بازه زمانی اول، سه محصول موجود است و محصول ۱ خریداری می‌شود. در بازه دوم محصولات ۱ و ۲ و ۵ موجود است و محصول ۵ خریداری می‌شود و به‌همین‌ترتیب. در این سناریو، سه نوع مشتری وجود دارد: $C_1 = \{1,2,3,4,5,0\}$، $C_2 = \{4,5,0\}$ و $C_3 = \{2,0\}$. در بازه اول، نوع مشتری C_1 سازگار در نظر گرفته می‌شود زیرا در صورت ورود ، محصول ۱ را خریداری میکند. در دوره دوم، فقط C_2 سازگار است زیرا محصول ۵ را خریداری می‌کند. در دوره چهارم، هیچ نوع مشتری سازگار نیست زیرا کسی وارد بازار نشده است. در دوره پنجم، انواع مشتریان C_2 و C_3 سازگار هستند، زیرا آنها هیچ محصولی را خریداری نمی‌کنند. اصطلاح «سازگاری» به این دلیل استفاده می‌شود که نمی‌توان به طور قطعی دانست که کدام نوع مشتری وارد یک بازه شده است. در جدول ۱ P، NP، NA به ترتیب مخفف خرید (Purchase)، عدم خرید (No-Purchase) و عدم ورود مشتری (No-Arrival) است.

جدول ۱. نمونه جدول بازه ای

	دوره				
محصولات	1	2	3	4	5
1	بله	بله	خیر	خیر	خیر
2	بله	بله	خیر	خیر	خیر
3	بله	خیر	خیر	خیر	بله
4	خیر	خیر	بله	خیر	خیر
5	خیر	بله	بله	بله	خیر
تراکنش انجام شده	1	5	4	NA	0
انواع مشتریان سازگار	{1}	{2}	{1,2 }	-	{2,3}
نوع تراکنش	T_P	T_P	T_P	T_{NA}	T_{NP}

مشتریان سازگار در دوره t با $M_t(j_t, S_t)$ مشخص می‌شود. برای مجموعه کامل انواع مشتریان $\{C_1, \dots, C_k\}$ و $j_t \in S_t$، که در آن S_t محصولات ارائه شده و j_t تراکنش است. مشتریان سازگار به صورت زیر نمایش داده میشود:

$$M_t(j_t, S_t) = \{i: C^{(i)}(j_t) < C^{(i)}(k) \ \ \forall k \in S_t, k \neq j_t\}$$

۳-۱- مجموعه‌داده استفاده شده

مجموعهٔ داده ای که استفاده شده، مربوط به پنج هتل Continental واقع در ایالات متحده است و عمدتاً شامل اطلاعات سفر افرادی است که برای اهداف تجاری اتاق رزرو کرده‌اند. این مجموعه‌داده به‌صورت عمومی در دسترس است و به‌عنوان بنچمارک برای مسئلهٔ کشف نوع مشتری استفاده می‌شود. بازه زمانی داده‌ها از ۱۲ مارس ۲۰۰۷ تا ۱۵ آوریل ۲۰۰۷ است که ۳۴ روز را شامل می‌شود [۲۵]. رزروها از طریق کانال‌های مختلفی مانند آژانس‌های مسافرتی، اپراتورهای هتل، بازدیدهای شخصی و وب‌سایت هتل‌ها انجام‌شده است. اتاق‌های هتل (محصولات) با حداقل زمان تحویل چهار هفته رزرو می‌شوند.

۳-۲- رویکرد الگوریتم ژنتیک

در این بخش، نگاهی اجمالی به رویکرد الگوریتم ژنتیک [۱۹] برای حل مسئلهٔ کشف نوع مشتری می‌پردازیم. این الگوریتم با ایجاد جمعیت اولیه با اندازهٔ ثابت شروع می‌شود. هریک از افراد جمعیت با لیستی از انواع مشتریان نشان داده می‌شود که به‌صورت تصادفی تولید می‌شود و می‌تواند اندازه تصادفی داشته باشد. در این رویکرد از دو تابع برازندگی مختلف استفاده می‌شود. تابع اول امکان سنجی راه حل (کروموزوم) را با تعیین تعداد دوره‌های زمانی که در آن هیچ نوع مشتری سازگار نیست، ارزیابی می‌کند. اگر راه‌حل سازگار تلقی شود، تابع اول مقدار برازندگی صفر را برمی‌گرداند و سپس تابع دوم اعمال می‌شود. تابع دوم مدل حداکثرسازی راست نمایی تخمین است که در فرمول (۱) آن را مشاهده می کنید.

$$\max_{x \geq 0} \mathcal{L}_l(x)$$

$$\text{s.t.} \sum_{i=1}^{k} x_i = 1 \qquad (۱)$$

که در آن x_i احتمال ورود مشتری نوع i را نشان می دهد. در اینجا، $L_l(x)$ تابع log-likelihood است که به صورت معادله (۲) بیان می‌شود. که در آن T_P دوره‌های خرید را نشان می‌دهد، T_{NP} دوره‌های بدون خرید و T_{NA} دوره‌های بدون ورود را نشان می‌دهد. بنابراین، $T = |T_P| + |T_{NP}| + |T_{NA}|$.

$$\mathcal{L}_l(X|\lambda) = \sum_{t \in T_P} \log\left(\sum_{i \epsilon M_t(j_t|S_t)} x_i\right) + \sum_{t \in T_{NA}} \log\left(\sum_{i \epsilon M_t(0|S_t)} x_i\right) + |T_{NA}| \log(1-\lambda) + (|T_P| + |T_{NP}|) \log \lambda \qquad (۲)$$

اولین عبارت در معادلهٔ (۲) نشان دهندهٔ محاسبهٔ احتمال log-likelihood تراکنش مشاهده‌شده در دوره‌های $t \in T_P$ است. جمله دوم محاسبه دوره های بدون خرید $t \in T_{NA}$ را نشان می دهد. عبارت سوم دوره های بدون ورود مشتری را نشان می دهد. معادله (۲) با x و λ قابل تفکیک است و به صورت سراسری در (x, λ) مقعر[۷] است. بنابراین، ما می‌توانیم $\lambda^* = \frac{|T_P| + |T_{NP}|}{T}$ پیدا کنیم.

۳-۳- رویکرد الگوریتم ممتیک

در این بخش، نگاهی اجمالی به رویکرد الگوریتم ممتیک [۳۴] جهت حل مسئلهٔ کشف نوع مشتری می‌پردازیم. الگوریتم‌های تکاملی خالص هنگامی که با تکنیک‌های دیگر مانند جست وجوی محلی ترکیب شوند، مؤثرتر هستند [۳۵]– [۳۶]. ممتیک ها نوعی الگوریتم هستند که راه‌حل ها را با ترکیب الگوریتم‌های تکاملی با جست وجوی محلی بهبود می‌بخشند [۷]. با استفاده از تابع برازندگی و پاسخ‌های پیداشده و وجود همسایگی، می توان راه‌حل های بهتری یافت و سریع تر به پاسخ بهینه رسید. شبه کد الگوریتم ممتیک استفاده‌شده در [۳۴] به شرح زیر است:

```
1  Generate random initial population
2  Local search
3  Fitness evaluation
4  Repeat until (termination condition)
5        Single-point crossover
         Reproduction operators
6        Local search
7        Fitness evaluation
8        Survival selection
9  End repeat
```

۴- رویکرد پیشنهادی

در این بخش، رویکرد پیشنهادی خود را برای افزایش همگرایی و دستیابی به یک راه‌حل قابل قبول (اولین جواب ممکن) در زمان سریع‌تر ارائه می‌کنیم.
جست وجوی الگوریتم فراابتکاری در فضای مسئله ای بزرگ می‌تواند زمان‌بر باشد و یافتن راه‌حل بهینه ممکن است همیشه امکان‌پذیر نباشد؛ بنابراین، برای به‌دست‌آوردن نتایج سریع‌تر، می‌توان رویکرد را بهبود بخشید تا شانس یافتن راه‌حل بهینه را افزایش دهد. الگوریتم‌های پیشنهادی در بخش‌های

قبلی از دوفاز تشکیل شده‌اند: فاز اولیه، پس از ایجاد جمعیتی تصادفی، نسل اولیه عمدتاً شامل راه‌حل‌های غیرقابل‌قبول هستند و نمی‌توان به‌عنوان راه‌حل آنها را پذیرفت. فاز ثانویه که ارزش برازندگی راه‌حل‌های قابل‌قبول را برای یافتن راه‌حل‌های بهینه بهبود می‌بخشد. این دوفاز را می‌توان با استفاده از جمعیت اولیهٔ بهتر به‌جای جمعیت تصادفی که هم گرایی فاز اولیه را سرعت می‌بخشد، بهبود داد. برای دستیابی به جمعیت اولیهٔ بهتر، انتخاب نوع‌های مشتری با تأثیرات قابل‌توجه بر تابع برازندگی بسیار مهم است. ایده اصلی این رویکرد پیشنهادی، استفاده از محصولات مؤثرتر برای ایجاد مجموعه راه‌حل است. برای رسیدن به این هدف، رویکردی سه مرحله ای پیشنهاد می‌کنیم. در مرحله اول، محصولات مؤثری را که احتمالاً توسط انواع مشتری خریداری می‌شود، شناسایی می‌کنیم. در مرحله دوم، با استفاده از محصولات مؤثر شناسایی‌شده، یک مجموعه راه‌حل ایجاد می‌کنیم. در مرحله سوم، بهترین راه‌حل‌ها را از مجموعه ساخته شده برای استفاده توسط الگوریتم‌های ژنتیک و ممتیک انتخاب می‌کنیم. هر نوع مشتری یک لیست ترجیحی از محصولات است و برازندگی یک راه‌حل مستقیماً تحت‌تأثیر انتخاب محصولات برای هر نوع مشتری است. راه حل شامل تعدادی نوع مشتری است و می‌توان تعداد دوره‌های زمانی ناسازگار را محاسبه کرد. گام اولیه برای یافتن راه‌حل بهینه، شناسایی محصولات مؤثر است؛ زیرا بر اثربخشی مشتریان تأثیر می‌گذارد. دو روش برای شناسایی محصولات مؤثر پیشنهادشده است، نخست روش میانگین که در آن میانگین فروش محصولات محاسبه می‌شود و محصولات با فروش بیشتر از میانگین به‌عنوان محصولات مؤثر انتخاب می‌شوند. دومین روش میانگین مختلط که شامل انتخاب ۸۰ درصد محصولات بالاتر از حد میانگین و ۲۰ درصد محصولات زیر میانگین به‌عنوان محصولات موثر است. در گام دوم یک مجموعه راه‌حل ایجاد میکنیم، در زیر شبه کد آن را مشاهده میکنید.

```
1 Find effective products e
2 For i in length(e)
3     all_perms.append(permutations(e, i))
4 Endfor
5 For j in range(length(e), length(n))
6     pool.append(combinatios(all_perms, j))
7 Endfor
```

پس از تولید مجموعه جواب، مرحله بعدی انتخاب از مجموعه جواب برای جمعیت اولیه است، برای این منظور می‌توان از سه روش را پیشنهاد داده‌ایم. روش اول انتخاب تصادفی است که در آن راه‌حل‌ها به طور تصادفی انتخاب می‌شوند تا جمعیت شروع الگوریتم را تشکیل دهند. برای مثال، اگر مجموعه راه‌حل‌ها [{a={1,3,2,0} , b={2,3,0} , c={3,0}] باشد، روش انتخاب تصادفی به طور تصادفی راه‌حل‌هایی را انتخاب می‌کند. روش دوم انتخاب بهترین راه‌حل است که در آن میانگین ارزش خرید همه محصولات محاسبه می‌شود و راه‌حل‌ها بر اساس این مقدار مرتب می‌شوند. ارزش خرید باتوجه‌به تعداد محصولات خریداری شده تعیین می‌شود. به‌عنوان‌مثال، اگر مجموعه راه‌حل [{a={1,3,2,0} , b={2,3,0} , c={3,0}] باشد و محصولات ۱، ۲، و ۳ به ترتیب به مقدار ۱۰، ۲۰ و ۳۰ فروخته شده باشند و تعداد خریدهای مشتری a برابر با ۶۰ و مشتری b برابر با ۵۰ و در آخر مشتری c برابر با ۳۰ است، همچنین میانگین ارزش خرید مشتریان به ترتیب ۲۰، ۲۵ و ۳۰ است. روش سوم مخلوط متوسط است که در آن ۸۰٪ از انتخاب ها از بهترین راه حل ها و ۲۰٪ باقی مانده از بدترین راه حل ها انجام می‌شود تا جمعیت شروع الگوریتم را تشکیل دهد.

۵- نتایج

برای ارزیابی کارایی روش پیشنهادی خود، آن را با جمعیت اولیهٔ تولیدشدهٔ کلاسیک (تصادفی) مقایسه کردیم. جدول (۲) نتایج به‌کارگیری جمعیت اولیهٔ روش پیشنهادی و کلاسیک در الگوریتم‌های ژنتیک و ممتیک را با اندازهٔ ثابت ۲۰ نشان می‌دهد. هر آزمایش ۳۰ بار تکرار شده و میانگین نتایج در جدول نمایش‌داده‌شده است. برازندگی بهترین راه‌حل در یک جمعیت به‌عنوان برازندگی جمعیت شناخته می‌شود. نتایج نشان می‌دهد درصورتی‌که جمعیت اولیه از روش‌های پیشنهادی ساخته‌شده باشد، هم الگوریتم ژنتیک و هم الگوریتم ممتیک اولین راه‌حل قابل قبول را با نسل‌های کمتری پیدا خواهند کرد؛ بنابراین روش پیشنهادی منجر به بهبود عملکرد در فاز اول و هم گرایی روش فراابتکاری و درنهایت، پیدا شدن سریع‌تر اولین جواب قابل قبول می‌شود. ستون اندازه، اندازهٔ اولین راه‌حل قابل قبول یافته شده توسط الگوریتم را نشان می‌دهد.

جدول ۲ - مقایسه جمعیت اولیه کلاسیک و روش‌های پیشنهادی (اندازه جمعیت ثابت ۲۰). ستون «رویکرد» به ترتیب استراتژی استفاده شده در مرحله اول و سوم رویکرد چند مرحله ای را نشان می دهد.

رویکرد		هتل ۱		هتل ۲		هتل ۳		هتل ۴		هتل ۵	
		تعداد نسل	اندازه	تعداد نسل	اندازه	تعداد نسل	اندازه	تعداد نسل	اندازه	تعداد نسل	اندازه
الگوریتم ژنتیک											
جمعیت اولیه کلاسیک		98.3	12.2	190	12.7	197	12.7	115.4	9.6	71.7	11.1
مرحله ۱	مرحله ۳										
میانگین	تصادفی	**60.4**	12.6	**157.1**	12.8	**72.5**	10.7	**63.2**	11.4	**35**	9.9
میانگین	میانگین	62.9	12.1	161.8	12.5	76.9	10.5	66.5	11.1	37	10.5
میانگین	مخلوط	63.7	12.7	164.3	12.9	80.6	10.6	67.4	10.9	38	10.9
مخلوط	تصادفی	64.7	12.1	164.5	12.6	80.1	10.9	67.9	9.9	38.3	10.4
مخلوط	میانگین	65.1	12.5	165.5	12.7	81.9	10.5	68.4	10.2	38.3	10.9
مخلوط	مخلوط	65.9	11.9	168.5	12.8	83.7	10.8	69.8	10.8	39.2	10.8

الگوریتم ممتیک											
جمعیت اولیه کلاسیک		74.1	12.8	124.5	11.8	157	13.3	87.9	10	59.8	10.6
مرحله ۱	مرحله ۳										
میانگین	تصادفی	**44.4**	12.6	**101.4**	11.2	**54.8**	10.9	**47.5**	9.8	**28.6**	9.7
میانگین	میانگین	46.2	12.7	104.6	11.6	59.1	11.3	50	9.5	30.4	10.3
میانگین	مخلوط	47.2	12.9	105.7	11.9	61.5	11.4	50.6	10.1	30.9	10.7
مخلوط	تصادفی	47.7	12.3	106.3	11.6	62	11.9	50.9	10.2	31	10.4
مخلوط	میانگین	48	12.5	107.4	11.4	62.7	10.8	51.8	9.9	31.4	10.3
مخلوط	مخلوط	48.8	12.6	108.7	11.3	64	11.8	52.8	10.3	32	10.2

در جدول (۳) رویکرد پیشنهادی سه‌مرحله‌ای را با روش کلاسیک در الگوریتم ژنتیک مقایسه کرده‌ایم. روش پیشنهادی با رویکرد میانگین و تصادفی بهترین عملکرد را داشته و منجر به کاهش تعداد نسل‌های موردنیاز برای یافتن اولین راه‌حل قابل‌قبول از ۱۷.۳٪ تا ۶۳.۳٪ در پنج مجموعه دادهٔ هتل نسبت به روش کلاسیک شده است. بیشترین کاهش در هتل ۳ با کاهش ۶۳.۳ درصدی مشاهده می شود.

جدول ۳ - مقایسه رویکرد سه مرحله ای پیشنهادی با کلاسیک در الگوریتم ژنتیک.

رویکرد		هتل ۱	هتل ۲	هتل ۳	هتل ۴	هتل ۵
مرحله ۱	مرحله ۳					
میانگین	تصادفی	**38.55%**	**17.31%**	**63.35%**	**45.23%**	**51.18%**
میانگین	میانگین	35.99%	14.83%	60.95%	42.33%	48.26%
میانگین	مخلوط	35.10%	13.50%	59.04%	41.56%	46.90%
مخلوط	تصادفی	34.11%	13.38%	59.33%	41.13%	46.58%
مخلوط	میانگین	33.73%	12.87%	58.41%	40.69%	46.47%
مخلوط	مخلوط	32.88%	11.30%	57.50%	39.48%	45.28%

جدول (۴) نشان می دهد که روش پیشنهادی با رویکرد میانگین و تصادفی بهترین عملکرد را داشته و در الگوریتم ممتیک منجر به کاهش ۱۸.۵٪ تا ۶۵٪ در تعداد نسل های مورد نیاز برای یافتن اولین راه حل قابل قبول در بین مجموعه داده های پنج هتل شده است. در این ارزیابی بیشترین کاهش در هتل ۳ با ۶۵٪ مشاهده شده است.

جدول ۴ - مقایسه روش پیشنهادی با کلاسیک در الگوریتم ممتیک.

رویکرد		هتل ۱	هتل ۲	هتل ۳	هتل ۴	هتل ۵
مرحله ۱	مرحله ۳					
میانگین	تصادفی	**40.09%**	**18.55%**	**65.06%**	**45.87%**	**52.13%**
میانگین	میانگین	37.57%	15.95%	62.30%	43.03%	49.11%
میانگین	مخلوط	36.31%	15.05%	60.82%	42.45%	48.32%
مخلوط	تصادفی	35.62%	14.58%	60.49%	42.01%	48.18%
مخلوط	میانگین	35.18%	13.72%	60.04%	41.03%	47.48%
مخلوط	مخلوط	34.12%	12.67%	59.22%	39.91%	46.42%

۶- نتیجه‌گیری

یافتن راه‌حل بهینه برای مسئله بهینه‌سازی می‌تواند کار چالش‌برانگیز باشد، به‌ویژه زمانی که به‌سرعت در پی راه‌حل‌های قابل‌قبول باشیم. درحالی‌که جمعیت‌های اولیه تصادفی اغلب در الگوریتم‌های فراابتکاری استفاده می‌شوند، استفاده از جمعیت غیر تصادفی برای تسریع فرایند الگوریتمی ممکن است بسته به کاربرد و فضای مشکل دشوار باشد. در این مطالعه، رویکردی سه مرحله ای برای رسیدگی به این چالش پیشنهاد کردیم: (۱) شناسایی محصولات مؤثر، (۲) تولید مجموعه‌ای از راه‌حل‌ها، و (۳) انتخاب راه‌حل‌ها برای جمعیت اولیه. ما این رویکرد را با اجرای الگوریتم ژنتیک و الگوریتم ممتیک در پنج مجموعه دادهٔ هتل برای حل مسئلهٔ «کشف نوع مشتری» آزمایش کردیم و نتایج نشان داد که استفاده از این رویکرد، عملکرد الگوریتم‌های ژنتیک و ممتیک را در یافتن راه‌حل‌های قابل‌قبول با تکرارهای کمتر بهبود می‌بخشد.

مراجع

[1] K. T. Talluri and G. J. Van Ryzin, *The Theory and Practice of Revenue Management*, vol. 68. Boston, MA: Springer US, 2004.Zachman, John A., *"A Framework for*

[19] M. HajMirzaei, K. Ziarati, and A. Nikseresht, "Discovering customer types using sales transactions and product availability data of 5 hotel datasets with genetic algorithm," *J. Revenue Pricing Manag.*, 2020, doi: 10.1057/s41272-020-00245-3.

[20] S. Jagabathula and G. Vulcano, "A Partial-order-based Model to Estimate Individual Preferences Using Panel Data," *SSRN Electron. J.*, no. April, 2017, doi: 10.2139/ssrn.2560994.

[21] H. Lee and Y. Eun, "Discovering heterogeneous consumer groups from sales transaction data," *Eur. J. Oper. Res.*, vol. 280, no. 1, pp. 338–350, Jan. 2020, doi: 10.1016/J.EJOR.2019.05.043.

[22] M. HajMirzaei, K. Ziarati, and A. Nikseresht, "A customer type discovery algorithm in hotel revenue management systems," *J. Revenue Pricing Manag.*, 2021, doi: 10.1057/s41272-020-00273-z.

[23] Toğan, V. and Daloğlu, A., 2008. "An improved genetic algorithm with initial population strategy and self-adaptive member grouping". Computers & Structures, 86(11-12), pp.1204-1218.

[24] Chevallier, M., Rogovschi, N., Boufarès, F., Grozavu, N., Clairmont, C. (2021). Seeding Initial Population, in Genetic Algorithm for Features Selection. In: , et al. Proceedings of the 12th International Conference on Soft Computing and Pattern Recognition (SoCPaR 2020). SoCPaR 2020. Advances in Intelligent Systems and Computing, vol 1383. Springer, Cham. https://doi.org/10.1007/978-3-030-73689-7_55

[25] T. Bodea, M. Ferguson, and L. Garrow, "Data Set —Choice-Based Revenue Management: Data from a Major Hotel Chain ," *Manuf. Serv. Oper. Manag.*, vol. 11, no. 2, pp. 356–361, 2008, doi: 10.1287/msom.1080.0231.

[26] Q. Li, S.-Y. Liu, and X.-S. Yang, 2020. "Influence of initialization on the performance of metaheuristic optimizers, " *Applied Soft Computing*, vol. 91, pp. 106193. doi: 10.1016/j.asoc.2020.106193

[27] B. Kazimipour, X. Li, and A. K. Qin, 2014, July in 2014 IEEE Congress on Evolutionary Computation (CEC) (pp. 2404-2411). IEEE. doi: 10.1109/CEC.2014.6900624.

[28] H. Deng, L. Peng, H. Zhang, B. Yang, and Z. Chen, 2019. Ranking-based biased learning swarm optimizer for large-scale optimization, Information Sciences, vol. 493, pp. 120-137. doi: 10.1016/j.ins.2019.04.037

[29] N. Henderson, M. de Sá Rêgo, J. Imbiriba, M. de Sá Rêgo, and W. F. Sacco, 2018. Testing the topographical global initialization strategy in the framework of an unconstrained optimization method, Optimization Letters, vol. 12, no. 4, pp. 727-741. doi: 10.1007/s11590-017-1137-6.

[30] Vlašić, M. Đurasević, and D. Jakobović, 2019. Improving genetic algorithm performance by population initialisation with dispatching rules, Computers & Industrial Engineering, vol. 137, pp. 106030. doi: 10.1016/j.cie.2019.106030.

[31] K. Łapa, K. Cpałka, A. Przybył, and K. Grzanek, 2018, June in International Conference on Artificial Intelligence and Soft Computing (pp. 449-461). Springer. doi: 10.1007/978-3-319-91253-0_42.

[32] M. Richards, and D. Ventura, 2004, July in IEEE Int. Joint. Conf. Neural (pp. 2309-2312). IEEE. doi: 10.1109/IJCNN.2004.1380986.

[33] B. Kazimipour, X. Li, and A. K. Qin, 2014, July in 2014 IEEE Congress on Evolutionary Computation (CEC) (pp. 2585-2592). IEEE. doi: 10.1109/CEC.2014.6900618.

[34] H. Sherafat Moula, H. Yaghoubyan, R. Malekhosseini and K. Bagherifard, "Customer type discovery in hotel revenue management by Memetic algorithm," *J. Revenue Pricing Manag.*, 2023, doi: 10.1057/s41272-022-00408-4

[35] L. Davis, Handbook of genetic algorithms. 1996.

[36] D. E. Goldberg and S. Voessner, "Optimizing global-local search hybrids.," in *GECCO*, 1999, vol. 99, pp. 220–228.

Information Systems Architecture", IBM Systems Journal, Vol. 26, No. 3, 1987.

[2] P. Liu and S. Smith, "Estimating unconstrained hotel demand based on censored booking data," *Journal of Revenue and ...*, July 01, 2002. http://link.springer.com/10.1057/palgrave.rpm.5170015 (accessed February 21, 2018).

[3] A. Nikseresht and K. Ziarati, "Estimating True Demand in Airline's Revenue Management Systems using Observed Sales," *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 7, pp. 361–369, 2017, doi: 10.14569/ijacsa.2017.080748.

[4] C. Y. Goh, C. Yan, and P. Jaillet, "Estimating Primary Demand in Bike-sharing Systems," *SSRN Electron. J.*, Jan. 2019, doi: 10.2139/ssrn.3311371.

[5] J. P. Newman, M. E. Ferguson, L. A. Garrow, and T. L. Jacobs, "Estimation of Choice-Based Models Using Sales Data from a Single Firm," *Manuf. Serv. Oper. Manag.*, vol. 16, no. 2, pp. 184–197, May 2014, doi: 10.1287/msom.2014.0475.

[6] P. Moscato, "On evolution, search, optimization, GAs and martial arts: toward memetic algorithms. California Inst. Technol., Pasadena," 1989.

[7] N. Krasnogor and J. Smith, "A tutorial for competent memetic algorithms: model, taxonomy, and design issues," IEEE Trans. Evol. Comput., vol. 9, no. 5, pp. 474–488, 2005.

[8] D. Bertsimas and V. V Mišic, "Data-driven assortment optimization," *Manage. Sci.*, vol. 1, pp. 1–35, 2015.

[9] S. Jagabathula, "Assortment Optimization Under General Choice," *Ssrn*, pp. 1–51, 2014, doi: 10.2139/ssrn.2512831.

[10] S. Jagabathula and P. Rusmevichientong, "A Nonparametric Joint Assortment and Price Choice Model," *Ssrn*, no. July, 2013, doi: 10.2139/ssrn.2286923.

[11] G. Gallego, H. Topaloglu, and others, *Revenue management and pricing analytics*, vol. 209. Springer, 2019.

[12] G. Bitran and R. Caldentey, "An overview of pricing models for revenue management," *Manuf. Serv. Oper. Manag.*, vol. 5, no. 3, pp. 203–229, 2003.

[13] S. Kunnumkal, "Randomization Approaches for Network Revenue Management with Customer Choice Behavior," *Prod. Oper. Manag.*, vol. 23, no. 9, pp. 1617–1633, Sep. 2014, doi: 10.1111/poms.12164.

[14] L. Chen and T. Homem-de-Mello, "Mathematical programming models for revenue management under customer choice," *Eur. J. Oper. Res.*, vol. 203, no. 2, pp. 294–305, Jun. 2010, doi: 10.1016/J.EJOR.2009.07.029.

[15] G. Vulcano, G. van Ryzin, and R. Ratliff, "Estimating Primary Demand for Substitutable Products from Sales Transaction Data," *Ssrn*, no. August 2015, 2011, doi: 10.2139/ssrn.1923711.

[16] G. van Ryzin and G. Vulcano, "A Market Discovery Algorithm to Estimate a General Class of Nonparametric Choice Models," *Manage. Sci.*, vol. 61, no. 2, pp. 281–300, 2015, doi: 10.1287/mnsc.2014.2040.

[17] G. van Ryzin and G. Vulcano, "Technical Note—An Expectation-Maximization Method to Estimate a Rank-Based Choice Model of Demand," *Oper. Res.*, vol. 65, no. 2, pp. 396–407, 2017, doi: 10.1287/opre.2016.1559.

[18] S. Jagabathula, D. Mitrofanov, and G. Vulcano, "Inferring Consideration Sets from Sales Transaction Data," *SSRN Electron. J.*, 2019, doi: 10.2139/ssrn.3410019.

۱ Maximum Likelihood Estimation Model
۲ Mixed Integer Program
۳ Expectation Maximization
۴ Choice-Based Demand Estimation
۵ Panel Data
۶ Linear Ordering Problem
۷ Concave

پنجمین کنفرانس ملی انفورماتیک ایران
پژوهشگاه دانشهای بنیادی، پردیس فرمانیه، تهران
۱۳ و ۱۴ دی ماه ۱۴۰۲

بهبود گراف زوم؛ چارچوبی برای یادگیری بازنمایی گراف

سید مهدی وحیدی پور[۱]، استادیار گروه هوش مصنوعی، عماد صلاتی*[۲]، دانشجوی کارشناسی ارشد مهندسی هوش مصنوعی، رسول سبزه‌واری[۲]، دانشجوی کارشناسی ارشد مهندسی نرم افزار ، محمدریاضی[۲]، دانشجوی کارشناسی ارشد مهندسی نرم افزار

[۱] گروه مهندسی کامپیوتر، دانشکده برق و کامپیوتر ـ دانشگاه کاشان ـ کاشان ـ ایران
[۲] کارشناسی ارشد، گروه مهندسی کامپیوتر، دانشکده برق و کامپیوتر ـ دانشگاه کاشان ـ کاشان ـ ایران

چکیده: بازنمایی گره‌های گراف با بردار امبدینگ می‌تواند کاربردهای مختلفی از یادگیری ماشین را در گراف تعریف کند؛ کاربردهایی مانند پیش‌بینی و رده‌بندی که در سطح گره و/یا یال بازتعریف می‌شوند. برای ایجاد بازنمایی برای گره‌های گراف می‌توان از روشهای سلسله مراتبی استفاده کرد. گراف‌زوم یکی از الگوریتم‌های یادگیری بازنمایی گره است که به صورت سلسله مراتبی کار می‌کند. ورودی گراف زوم یک گراف وزن‌دار است. پس باید گراف ورودی وزن دار شود. در روش اصلی برای این قسمت، تنها از اطلاعات محتوایی گره‌ها استفاده می‌شود (فرض می‌شود که در هر گره محتوایی وجود دارد و هر چقدر دو گره محتوایی شبیه‌تر داشته باشند، باید وزن بین آن دو بیشتر شود). در این مقاله، اطلاعات ساختاری نیز در نظر گرفته می‌شود تا روشهای مختلفی برای وزن دار کردن گراف پیشنهاد شود (هر چقدر مسیرهای ارتباطی میان دو گره در گراف بیشتر باشد، وزن روی یال بین آن دو بیشتر می‌شود). روشهای پیشنهادی این مقاله منجر به تولید سه نسخه متفاوت از گراف زوم می‌شود: ساخت گراف وزن‌دار تنها با استفاده از اطلاعات محتوایی، ساخت گراف وزن‌دار تنها با استفاده از اطلاعات ساختاری و ساخت گراف وزن‌دار با استفاده از یک ترکیب خطی از اطلاعات ساختاری و محتوایی. آزمایشهای انجام شده نشان می‌دهد که استفاده از اطلاعات ساختاری منجر به بهبود جزیی معیارهای کارایی می‌شوند. اما نکته جالب آن است که استفاده از ترکیب اطلاعات ساختاری و محتوایی در وزن‌دار کردن گراف ورودی منجر به افزایش سرعت قابل توجه در گراف زوم می‌شود؛ حداقل ۲۴ درصد در تمامی آزمایش‌ها.

واژه‌های کلیدی: یادگیری ماشین در گراف، گراف زوم، امبدینگ گره، بازنمایی سلسله‌مراتبی، پیش‌بینی پیوند.

* سید مهدی وحیدی پور، **vahidipour@kashanu.ac.ir**

• مقدمه

در سال‌های اخیر کاربرد امبدینگ[1] گره‌ها در روش‌های تحلیل گراف افزایش یافته است؛ امبدینگ یک گره، یک بردار عددی است که تعداد عناصر آن به نسبت تعداد کل گره‌های گراف بسیار کمتر است. به عبارت دیگر، امبدینگ هر گره، نقطه‌ای را در فضای برداری (یا فضای امبدینگ) برای آن مشخص می‌کند و یا گره را در فضایی جدید بازنمایی[2] می‌کند. روش‌های مختلفی برای ساخت امبدینگ‌ها وجود دارد.

در ساخت امبدینگ باید ساختار[3] گراف حفظ شود. در ساختار گراف مشخص می‌شود کدام گره‌ها با هم مرتبط هستند و همسایه محسوب می‌شوند. حفظ ساختار به معنی آن است که بردار امبدینگ گره‌های همسایه به همدیگر شبیه باشد. بنابراین، ساخت امبدینگ یک مساله بهینه‌سازی است که در آن بردارهای امبدینگ برای هر گره به‌دست می‌آید به گونه‌ای که بردار امبدینگ گره‌های همسایه (نقاط مربوط به گره‌ها در فضای امبدینگ) به یکدیگر شبیه (نزدیک هم) و نقاط متناظر با گره‌های غیرهمسایه از همدیگر دور باشند.

برای حل مسئله بهینه‌سازی ساخت امبدینگ، می‌توان از روش‌های یادگیری استفاده کرد که به یادگیری بازنمایی گراف[4] معروف هستند. این روش‌ها به طور کلی به دو دسته روش‌های کم‌عمق[5] و روش‌های شبکه عصبی گرافی[6] تقسیم می‌شوند [۱]. روش‌های کم‌عمق، از تکنیک‌های جبر خطی، برای کاهش ابعاد گراف و در عین حال حفظ خواص آن استفاده می‌کنند [۲]. در حالی که روش‌های شبکه عصبی گرافی شامل شبکه‌های عصبی طراحی‌شده برای عملکرد مستقیم بر روی ساختار گراف، مانند شبکه‌های پیچشی گرافی[7] است [۳]. بر خلاف روش‌های کم‌عمق، روش‌های شبکه عصبی به تکنیک‌های جداگانه برای پیش پردازش داده‌ها مانند تجزیه و تحلیل شبکه، الگوریتم‌های تشخیص الگو و ساختار، تجزیه و تحلیل طیفی و غیره نیاز ندارند.

برای بدست آوردن امبدینگ گره می‌توان از نگاه سلسله مراتبی نیز استفاده نمود[8]. گراف زوم یکی از روشهایی است که در آن از نگاه سلسله‌مراتبی استفاده شده است [٤]. در گراف زوم قبل از آنکه برای گره‌ها امبدینگ محاسبه شود، گراف فشرده می‌شود. بر اساس گراف فشرده (که تعداد گره کمتری دارد) امبدینگ استخراج می‌شود. امبدینگ استخراج شده برای یک گره (در گراف فشرده)، با اندکی اصلاح به عنوان امبدینگ برای تعدادی از گره‌های گراف اولیه در نظر گرفته می‌شود.

البته گراف زوم روی گراف وزن‌دار انجام می‌شود. بنابراین، گراف اصلی باید در ابتدا به یک گراف وزن‌دار تبدیل شود. در این مقاله، برای وزن‌دار کردن گراف روشهای مختلفی پیشنهاد شده است. در این روشها از اطلاعات ساختاری (ارتباط میان گره‌ها با نماد S)، اطلاعات محتوایی (ویژگیها و اطلاعات مربوط به گره‌ها با نماد C) و ترکیب اطلاعات ساختاری و محتوایی (با نماد SC) استفاده می‌شود؛ که به ترتیب باعث ایجاد نسخه‌های GraphZoom_S، GraphZoom_C و GraphZoom_SC می‌شود.

نسخه‌های متفاوت ایجاد شده از گراف زوم، در بخش آزمایش‌ها با یکدیگر مقایسه می‌شوند. نتایج نشان می‌دهد که استفاده از اطلاعات محتوایی و ساختاری در وزن‌دار کردن گراف اولیه به طور متوسط، سرعت اجرای گراف زوم را حداقل ۲٤ درصد بهبود می‌دهد.

ادامه ساختار مقاله به صورت زیر است. در بخش دوم، مفاهیم پایه نظیر امبدینگ و مشابهت ساختاری و محتوایی مرور می‌شوند. بخش سوم، چارچوب گراف زوم و مراحل آنرا معرفی می‌کند. بخش چهارم روش‌های پیشنهادی را بیان می‌کند. آزمایش‌های انجام شده روی روشهای پیشنهادی و نتایج حاصل

از آن در بخش پنجم بررسی شده است. درنهایت، بخش ششم به نتیجه‌گیری اختصاص دارد.

۲ مفاهیم پایه

در این بخش مفاهیم اصلی و مورد نیاز مانند امبدینگ گراف و مشابهت درگراف توضیح داده می‌شود .در این راستا، مفاهیمی نظیر نمایش گراف‌ها در ماتریس، مشابهت در گراف و امبدینگ گراف، معرفی می‌شوند.

۲-۱ گراف‌ها و نمایش آنها در ماتریس‌ها

گراف مجموعه‌ای از نقاط (یا گره‌ها) و اتصالات (یا یال‌ها) بین آنهاست. گره‌ها می‌توانند هر چیزی را نشان دهند، به عنوان مثال، افراد در یک شبکه اجتماعی، شهرها در نقشه، یا صفحات وب در یک وب سایت. یال‌هاها نحوه اتصال گره‌ها را نشان می‌دهند، به عنوان مثال، دوستی بین افراد، جاده‌های بین شهرها یا پیوندهای بین صفحات وب. درواقع، گراف G به صورت $G(V,E)$ تعریف می‌شود، که در آن V مجموعه گره‌ها و E مجموعه یال‌ها است. یال‌های E نشان‌دهنده روابط بین گره‌ها در V هستند. برای مثال، اگر گره‌های v و w را در V داشته باشیم که بین آنها رابطه دارند، آنگاه یک یال $e = (v, w)$ در E وجود دارد که آنها را به هم متصل می‌کند.

برای محاسبات راحت‌تر روی داده‌های گرافی، نیاز است تا آن‌ها را در ماتریس‌ها ذخیره کنیم. این کار به ما این امکان را می‌دهد که گراف را به صورت کارآمد در الگوریتم‌های کامپیوتری نمایش و کنترل کنیم.

ماتریس مجاورت[۹] (A) یک نمایش از یک گراف به صورت ماتریسی است. این یک ماتریس مربعی با اندازه $n \times n$ است که در آن n تعداد گره‌های گراف است. اگر گره i به گره j متصل باشد، می گوییم که یک یال از i تا j وجود دارد و درایه مربوطه $A[i][j]$ در ماتریس ۱ است. اگر یال وجود نداشته باشد، درایه ۰ است. اگر گراف دارای وزن باشد، ورودی در ماتریس مجاورت می‌تواند به صورت وزن یال تنظیم شود. برای یک گراف بدون جهت، ماتریس مجاورت قابلیت تقارن دارد که به این معناست که ورودی در ردیف i و ستون j ماتریس برابر با ورودی در ردیف j و ستون i ماتریس است [۵].

ماتریس درجه[۱۰] (D) یک ماتریس مورب است که در آن ورودی مورب $D[i][i]$ نشان‌دهنده تعداد یال‌هایی است که به گره i متصل است. یک ماتریس لاپلاسین[۱۱] (L) به صورت $L = D - A$ تعریف می‌شود که D ماتریس درجه و A ماتریس مجاورت است. ماتریس لاپلاسین در الگوریتم‌های گراف مختلف مانند خوشه‌بندی طیفی و پردازش سیگنال گراف استفاده می‌شود. با استفاده از ماتریس لاپلاسین می‌توان گراف‌ها را به خوشه‌های مختلف تقسیم‌بندی کرد. برای مثال در شبکه‌های اجتماعی، با استفاده از ماتریس لاپلاسین می‌توان کاربران را به خوشه‌هایی که بر اساس سلیقه، رفتار و علایقشان تفکیک شده‌اند، تقسیم‌بندی کرد [٦].

ماتریس ویژگی X شامل اطلاعات و ویژگیهای گره‌های گراف است. این یک ماتریس $n \times m$ است که در سطر i بردار ویژگی با m بُعد برای گره i وجود دارد. ماتریس‌های ویژگی را می‌توان در الگوریتم‌های گراف مختلف مانند طبقه‌بندی گره و تشخیص جامعه استفاده کرد [٦].

۲-۲ مشابهت در گراف

روش‌های شباهت به مجموعه‌ای از تکنیک‌های تحلیل گفته می‌شود که برای اندازه‌گیری شباهت یا نزدیکی بین گره‌های مختلف در یک گراف استفاده می‌شود. این روش‌ها به‌طور معمول برای وظایفی مانند پیش‌بینی لینک‌ها[۱۲]، شناسایی اجتماعات[۱۳] و سیستم‌های توصیه‌گر[۱٤] مورد استفاده قرار می‌گیرند. در ادامه روش‌هایی که در مقاله استفاده شده، بیان می‌شوند [۷].

۲-۲-۱ مشابهت ساختاری

مشابهت ساختاری در گراف، اندازه‌گیری شباهت دو گره مختلف بر اساس خواص ساختاری آن‌ها است؛ منظور از ساختار نحوه اتصالات دو گره و همسایگان آنها است [۸]. روش‌های مختلفی برای محاسبه تشابه ساختاری بین دو گره وجود دارد که در ادامه معیار مشابهت آدامیک-آدار[۱۵] توضیح داده می‌شود.

معیار آدامیک-آدار، برای محاسبه شباهت بین دو گره در یک گراف، تعداد همسایگان مشترک با هر یک از آن‌ها را محاسبه می‌کند. برای محاسبه تشابه آدامیک-آدار می‌توان با شمارش تعداد همسایه‌های مشترک بین دو گره i و j و با گرفتن معکوس لگاریتم امتیاز وزن‌دار سطح دو همسایه به فرمول زیر دست می‌یابد.

$$AA(i,j) = \sum_{k \in N_i \cap N_j} \left(\frac{1}{\log\left(D[k][k]\right)}\right) \quad (۱)$$

که در این فرمول، k همسایه مشترک بین گره i و j، $D[k][k]$ درجه گره k است که از ماتریس D استخراج شده است و در نهایت $AA(i,j)$ مشابهت آدامیک-آدار بین بین گره i و j است.

۲-۲-۲ k-نزدیک‌ترین همسایه[۱۶]

الگوریتم k-نزدیکترین همسایه، یک نوع از الگوریتم‌های یادگیری ماشین[۱۷] است که برای حل مسائل دسته‌بندی و رگرسیون استفاده می‌شود.

در یادگیری مبتنی بر گراف، هدف برچسب‌گذاری گره‌های نامشخص گراف بر اساس برچسب‌های گره‌های مشخص شده است. KNN برای یافتن k نزدیک‌ترین همسایه یک گره‌ی نامشخص بر اساس ساختار گراف استفاده می‌شود. سپس برچسب این همسایه‌های نزدیک استفاده می‌شود تا برچسب گره‌ی نامشخص به دست آید.

برای پیاده‌سازی KNN در یادگیری مبتنی بر گراف، گراف به عنوان یک ماتریس مجاورت یا یک ماتریس گراف لاپلاسین نمایش داده می‌شود. الگوریتم KNN روی گراف اجرا می‌شود تا k نزدیک‌ترین همسایه‌ی هر گره‌ی نامنظم شناسایی شود.

۲-۲-۳ مشابهت محتوایی

مشابهت محتوایی به اندازه‌گیری شباهت بین محتوای دو متن، تصویر، ویدئو یا هر محتوای دیگری با هدف پردازش زبان طبیعی[۱۸] یا درک تصاویر و ویدئو را گفته می‌شود [۸]. روشی که در این مقاله استفاده شده روش مشابهت کسینوسی[۱۹] می‌باشد.

مشابهت کسینوسی به یکی از روش‌های محاسبه شباهت بین دو بردار در یک فضای چند بعدی گفته می‌شود. در پردازش زبان طبیعی، ما می‌توانیم از مشابهت کسینوسی برای مقایسه دو متن با هم استفاده کنیم، برای محاسبه تشابه کسینوسی، ابتدا باید هر متن را به یک بردار تبدیل کنیم.

بعد از ایجاد بردار برای هر دو گره، می توانیم تشابه کسینوسی بین آن‌ها را با استفاده از فرمول زیر محاسبه کنیم:

$$(A,B) = \frac{(A.B)}{(\|A\|.\|B\|)} \quad (۲)$$

در این معادله، A و B دو برداری هستند که مورد مقایسه قرار می‌گیرند، نماد "·" به ضرب نقطه‌ای و نماد "||" به مقدار اندازه (نُرم) بردار اشاره دارد.

به طور خلاصه، شباهت کسینوسی یکی از روش‌های استفاده شده به عنوان معیاری برای اندازه‌گیری شباهت بین دو بردار است که اعدادی در بازه بین ۱- تا ۱ را می‌دهد و مقدار مشابهت نزدیک به ۱، بدین معنی است که دو بردار داده شده نزدیک به هم هستند. در زمینه پردازش زبان طبیعی، مقادیر نزدیک به ۱ به معنی شباهت بیشتر بین دو متن و مقادیر نزدیک به صفر بیانگر عدم شباهت بین دو متن هستند.

۲-۳ امبدینگ گره

امبدینگ یک گره آرایه ای به طول d است که می‌توان از آن به عنوان مختصات یک نقطه در فضای d بُعدی یاد کرد. این نقطه در فضای d بُعدی امبدینگ معادل گره در فضای گراف است. به طور خاص، یک گراف $G(V,E)$ دارای یک تابع امبدینگ $f: V \to \mathbb{R}^d$ است به طوری که به ازای هر گره یک نقطه را در

فضای امبدینگ مشخص می‌کند. این تابع باید یاد بگیرد چگونه گره‌های همسایه در گراف را به نقاط نزدیک به هم در فضای امبدینگ نگاشت کند [۹].

امبدینگ گره یک مسئله بهینه سازی محسوب می‌شود که با روش‌های مختلف قابل پیاده‌سازی است. در ادامه دو الگوریتم Node2Vec و DeepWalk مرور می‌شوند. در این دو روش، برای برای گره i مجموعه‌ای از گره‌ها ($N_R(i)$) مشخص می‌شود که در اطراف گره i قرار دارند. مسئله بهینه‌سازی برای یافتن امبدینگ گره i به این نکته توجه دارد که امبدینگ این گره باید نزدیک امبدینگ گره‌های داخل مجموعه $N_R(i)$ قرار بگیرد. اگر این موضوع برای تمامی گره‌ها رعایت شود، تابع f موفق به یادگیری نگاشت بین فضای گراف و فضای امبدینگ شده است.

برای یافتن مجموعه گره‌های اطراف گره i، یعنی $N_R(i)$، می‌توان از قدم‌زنی تصادفی استفاده کرد. قدم‌زنی تصادفی به معنای پیمایش گره‌ها در گراف با استراتژی R است. در روش DeepWalk و Node2Vec استراتژی های متفاوتی وجود دارد.

۲-۳-۱ DeepWalk

در گراف بدون جهت $G(V,E)$، هدف این روش یادگیری تابع $f: V \to \mathbb{R}^d$ است که هر گره v را به یک بردار d بعدی نگاشت می‌کند. الگوریتم به صورت زیر کار می‌کند:

- به طور تصادفی، تعداد زیادی پیمایش تصادفی[۲۰] کوتاه را از گراف نمونه‌برداری می‌کند. پیاده‌روی تصادفی، دنباله‌ای از گام‌های برداشته‌شده در گراف است که در آن هر گام به‌طور تصادفی از مجموعه گره‌های همسایه انتخاب می‌شود. گره‌هایی که روی یک پیاده‌روی تصادفی قرار دارند باید امبدینگ‌های شبیه به هم داشته باشند.
- هر پیمایش تصادفی مانند یک جمله است و هر گره معادل یک کلمه. بنابراین می‌توان از مدلهای زبانی مانند اسکیپ‌گرام استفاده کرد تا برای هر کلمه (یا همان گره) امبدینگ مناسب را پیدا کند. این مدل زبانی می‌تواند برای کلماتی که در یک رشته قرار دارند، امبدینگ شبیه پیدا کند. این همان چیزی است که در امبدینگ گره مورد انتظار است.

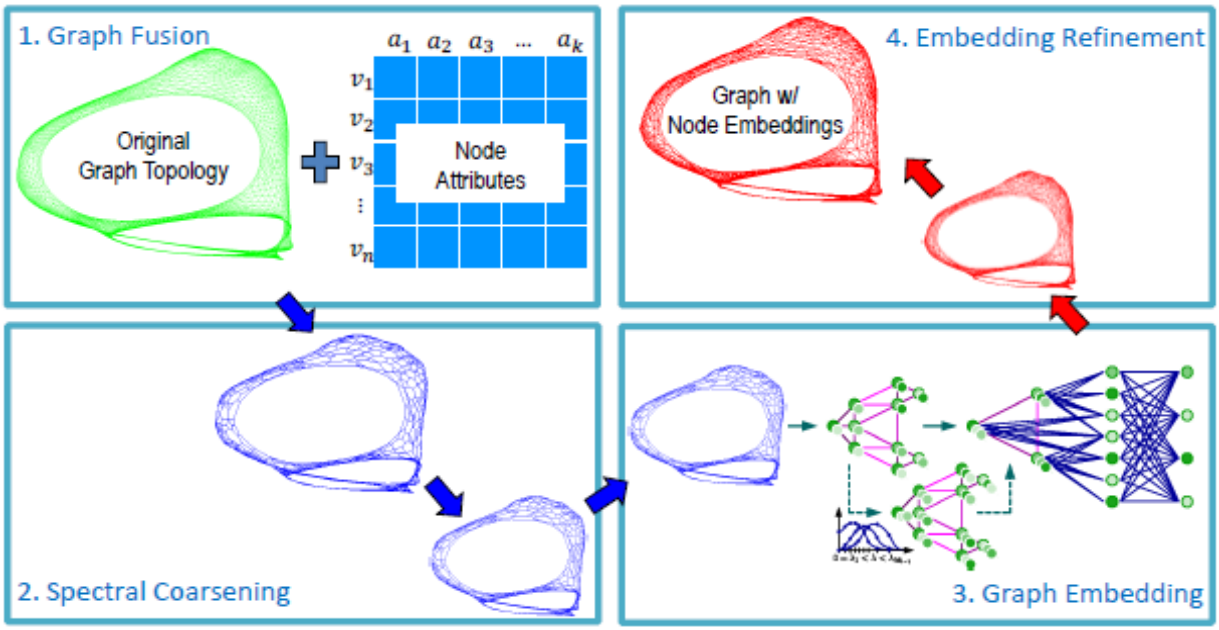
۲-۳-۲ Node2Vec

Node2Vec تعمیم از الگوریتم دیپ‌واک[۲۱] است که امکان کنترل بیشتر بر روی انواع پیمایش‌های تصادفی ایجاد شده را فراهم می‌کند. به طور خاص، از یک روش پیاده‌روی تصادفی مغرضانه[۲۲] استفاده می‌کند که بین جستجوی عرضی[۲۳] و جستجوی اول عمقی[۲۴] در گراف تعادل برقرار می‌کند. هدف Node2Vec همان دیپ‌واک است: تابع $f: V \to \mathbb{R}^d$ را یاد بگیرد که هر گره v به یک بردار d بعدی $f(v)$ نگاشت شود. الگوریتم Node2Vec از پیمایش‌های تصادفی روی گراف از هر گره v استفاده می‌کند. در طول یک پیمایش تصادفی، در هر مرحله، الگوریتم گره بعدی را برای بازدید بر اساس احتمال انتقال انتخاب می‌کند که به شباهت ساختاری بین گره‌ها بستگی دارد.

این احتمالات انتقال بر اساس دو متغیر p و q تعریف می‌شوند. متغیر p احتمال بازدید مجدد از گره‌ای را که قبلاً بازدید شده است را کنترل می‌کند، در حالی که متغیر q احتمال کاوش یک گره جدید در گراف را کنترل می‌کند.

با تنظیم مقادیر p و q، می‌توانیم کاوش و بهره‌برداری از ساختار گراف را متعادل کنیم، که به استخراج ساختارهای محلی و کلی از گراف کمک می‌کند.

در نهایت، الگوریتم Node2Vec از یک مدل، برای آموزش نمایش‌های برداری گره‌ها بر اساس پیمایش‌های تصادفی تولید شده از گراف استفاده می‌کند.



شکل ۱: نمای کلی چارچوب گراف‌زوم [۴]

۳ چارچوب گراف زوم

از آنجا که نوآوری این مقاله روی روش گراف زوم است، در این بخش گراف زوم با جزییات بیشتر شرح داده می‌شود. شکل (۱)، این چارچوب را نشان می‌دهد که از چهار فاز کلیدی تشکیل شده است:

- فاز (۱) گراف فیوژن[۲۵] که ویژگی‌های گره و اطلاعات ساختاری گراف اصلی را برای ساخت یک گراف وزنی ترکیب می‌کند.
- در فاز (۲)، با استفاده از روش اسپکترال کورسنینگ[۲۶]، گراف فشرده‌تر می‌شود.
- در فاز (۳)، هر یک از روش‌های گراف امبدینگ، مانند Node2Vec، روی گراف فشرده شده مرحله قبل اعمال می‌شود.
- در فاز (٤)، امبدینگ بدست آمده برای یک گره در گراف فشرده برای تعدادی از گره‌ها در گراف اصلی لحاظ می‌شود. این امبدینگ تا حدودی اصلاح می‌شود تا به دقت بیشتری در امبدینگ نهایی برسیم. این اصلاح با اضافه کردن اطلاعات اضافی به ساختار گراف انجام می‌شود [٤].

۳-۲ گراف فیوژن

هدف اصلی این گراف، ساخت یک گراف وزندار بر اساس توپولوژی اصلی گراف و اطلاعات ویژگی گره‌های آن است. این گراف وزندار، دارای همان تعداد گره‌های اصلی است اما دارای یال‌های وزن دار است که تعداد یالها می‌تواند با گراف اصلی متفاوت باشد [۱۰]. برای دستیابی به گراف وزندار، از تابع $fusion()$ استفاده می‌شود که ماتریس مجاورت گراف اصلی A_{topo} و ماتریس ویژگی گره‌ها X را به عنوان ورودی می‌گیرد و خروجی آن یک گراف وزندار G_{fusion} است.

تابع $fusion$ برای ایجاد گراف وزن‌دار، نیاز به دو گراف دارد: گراف اصلی G_{topo} (یا توپولوژی) و گراف ویژگی G_{feat}. از ترکیب این دو گراف، گراف وزن‌دار فیوژن تولید می‌شود.

گره‌های گراف ویژگی همان گره‌های گراف اصلی است. اما یالهای میان گره‌ها به صورت متفاوتی ترسیم می‌شود؛ هر گره به k نزدیکترین همسایه خود بر اساس ویژگیهای موجود در ماتریس X وصل می‌شود [۱۱]. با این حال، اجرای ساده مقایسه همه جفت‌های گره ممکن، در بدترین حالت دارای پیچیدگی زمانی $O(N^2)$ است که برای گراف‌های بزرگ مقیاس‌پذیر نیست. برای غلبه بر این محدودیت، از یک روش دسته‌بندی گره‌ها استفاده می‌شود؛ مانند روش دسته‌بندی اسپکترال. بدین ترتیب هر گره به k نزدیکترین همسایه در دسته خود وصل می‌شود. بدین ترتیب، یاقتن نزدیکترین همسایه نیازمند $O(M^2)$ عملیات است که در آن، M میانگین تعداد گره‌ها در هر خوشه است. از آنجا که تقریباً $\frac{N}{M}$ خوشه موجود است، پیچیدگی زمانی برای ساخت گراف ویژگی، برابر با $O(MN)$ است. در نتیجه، پیچیدگی زمانی کلی، خطی می‌شود.

پس از تشکیل گراف ویژگی، بر اساس شباهت محتوایی بین بردارهای ویژگی دو گره تصادفی، به هر یال، وزنی اختصاص داده می‌شود. در نهایت، گراف توپولوژی و گراف ویژگی با استفاده از یک جمع وزنی ترکیب شده و گراف فیوژن به دست می‌آید.

۳-۳ اسپکترال کورسنینگ

هدف فاز دوم گراف زوم، کاهش اندازه گراف اولیه با حفظ خصوصیات مهم آن است. یکی از روش‌های دستیابی به این هدف، استفاده از تکنیک گراف اسپکترال است که شامل امبدینگ گراف در فضای k-بعدی با استفاده از اولین k بردار ویژه[۲۷] از ماتریس لاپلاسین گراف است. با این حال، محاسبه بردار ویژه گراف‌های بزرگ هزینه محاسباتی بالایی دارد [۱۲].

برای غلبه بر این چالش، گراف زوم از یک رویکرد امبدینگ اسپکترال محلی کارآمد و مؤثر استفاده می‌کند. این رویکرد بر اساس تکنیک‌های پردازش سیگنال گراف، خوشه‌های گره را مشخص می‌کند [۱۳]. در این روش، به جای استفاده مستقیم از چند بردار ویژه اول گراف لاپلاسین اصلی، با اعمال فیلتر پایین گذر گراف[۲۸] به k بردار تصادفی، بردارهای هموار[۲۹] شده برای امبدینگ گراف k بعدی را می‌توان در زمان خطی به دست آورد. این طرح که به صورتی عصاره و چکیده گراف اصلی را تجمیع می‌کند، بر اساس محاسبات مبتنی بر قرابت و نزدیکی گره‌ها، خواص ساختاری گراف اصلی را حفظ می‌کند. در نهایت، استفاده از تکنیک‌های امبدینگ اکسپکترال محلی و عملگرهای نگاشت، آن را به ابزاری قدرتمند برای برنامه‌های امبدینگ گراف تبدیل می‌کند.

۴-۳ امبدینگ گراف

مرحله امبدینگ گراف، یکی از مراحل کلیدی در چارچوب گراف زوم است و شامل به دست آوردن امبدینگ گره در گراف کورسنینگ شده Gl است. بدین منظور، می‌توان از هر روش امبدینگ بدون نظارت که با تابع $f()$ نشان داده می‌شود استفاده نمود. هدف این روش‌ها ایجاد امبدینگ‌های با کیفیت بالا با ثبت ویژگی‌های مربوطه در گراف است [۱۴].

هنگامی که گراف کورسنینگ شده G_l ساخته شد، می‌توان از روش‌های امبدینگ نظارت نشده برای به دست آوردن امبدینگ گره‌های E_l روی G_l استفاده کرد. فرمول این فرآیند به صورت $E_l = f(G_l)$ است، که در آن، E_l نشان‌دهنده امبدینگ گره‌ها و $f()$ روش امبدینگ بدون نظارت مورد استفاده را نشان می‌دهد.

روش‌های امبدینگ بدون نظارت مختلفی وجود دارد که می‌توانند در این مرحله استفاده شوند، مانند DeepWalk[۳۰] یا Node2Vec. این روش‌ها از استراتژی‌های متفاوتی برای به

تصـویر کشـیدن سـاختار و روابط زیربنایی در گراف اسـتفاده می‌کنند، اما هدف همه آن‌ها ایجاد امبدینگ‌های گره با کیفیت بالا است.

به طور خلاصـه، فاز امبدینگ گراف در گراف زوم، با اسـتفاده از روش‌های امبدینگ نظارت نشـده، گره‌های یک گراف را بر روی یک فضای کم‌بعد پیوسته ترسیم می‌کند. همچنین، با حفظ روابط سـاختاری گره‌های گراف، پردازش کارآمد و دقیق گراف‌های مقیاس بزرگ را امکان‌پذیر می‌کند.

۳-۵ اصلاح امبدینگ

فاز نهایی، اصـلاح اسـت که برای اطمینان از دقت در نتایج امبدینگ بسیار مهم است. در طول این فاز، اطلاعاتی اضـافی در ساختار گراف گنجانده می‌شود تا امبدینگ‌ها را اصلاح کند. این اطلاعات اضـافی می‌تواند به شـکل گره‌های برچسـب‌دار، لبه‌های اضـافی یا سـایر ویژگی‌های گراف سـاختاری باشـد. با در نظر گرفتن این اطلاعات اضـافی، امبدینگ‌ها را می‌توان برای نمایش دقیق ساختار گراف اصلاح و بهینه کرد.

مرحله اصـلاح امبدینگ به ویژه برای برنامه‌هایی که به دقت بالایی در امبدینگ گراف خود نیاز دارند، مهم اسـت [۱۵]. به عنوان مثال، در زمینه سیستم‌های توصیه‌گر، داشتن امبدینگ‌های دقیق می‌تواند توانایی سیسـتم را در ارائه توصـیه‌های شـخصـی بهبود بخشد.

٤ روش‌های پیشنهادی

در این پژوهش، سـه نسـخه متفاوت از گراف زوم با نامهای GraphZoom_S، GraphZoom_C و GraphZoom_SC پیشنهاد شده است. در تمامی نسخه‌ها، فاز اول یعنی فیوژن تغییر کرده اسـت. فیوژن بر اسـاس معیارهای مختلف سـاختاری و/یا محتوایی انجام می‌شـود. در ادامه فیوژن موجود در هر نسـخه توضیح داده می‌شوند. بدیهی است سایر فازهای الگوریتم، مطابق با چارچوب اصلی گراف زوم است [٤].

٤-۱ GraphZoom_C

این نسـخه از گراف زوم همان نسـخه اصـلی الگوریتم [٤] است؛ فیوژن در این نسـخه تنها از اطلاعات محتوایی برای سـاخت گراف ویژگی استفاده می‌کند. شبه کد 1، فیوژن را در این نسخه نشان می‌دهد.

بنابر شـبه کد (۱)، بین تمامی جفت گره‌های یک خوشـه (مانند $(v_1, v_2 \in S_j$) مشـابهت محتوایی محاسبه می‌شـود.. سپس برای هر گره v، تعداد k نزدیکترین همسـایه انتخاب و بین آنها یال برقرار می‌شـود.. یالهای بدسـت آمده ماتریس E' را می‌سـازد (یالهای گراف G_{feat} است). دو گراف محتوا G_{feat} و ساختار G_{topo} با هم فیوژن می‌شـوند که نحوه انجام آن دقیقا معادل الگوریتم اصلی گراف زوم [٤] است.

Fusion ($G_{topo} = (V, E), X, m, k$)

Input: G_{topo} is the original graph, X is the feature matrix, m is the number of clusters, and k is the number of the nearest neighbors.
output: G

1. $G_{feat} = (V, E' = \{\}, W)$
2. r=$\frac{|V|}{m}$
4. $\{S_1, \dots, S_r\}$= Cluster G using a method
5. **For** each cluster S_j
6. **For** $v_1, v_2 \in S_j$ v_2
7. $T_E(v_1, v_2)$=Cosine_similarity (v_1, v_2,X)
8. **For** $v \in V$
9. Select kNN based on T_E and construct E'
10. Assign weigh to each edge in E' [4].
11. Fuse G_{topo} and G_{feat} and generate G[4].

def Cosine_similarity(v_1, v_2, X):
12. $dot\_ = abs\ (np.dot(v_1, v_2))$
13. $norm_{v_1} = LA.norm(v_1)$
14. $norm_{v_2} = LA.norm(v_2)$
15. **If** $norm_{v_1}$ or $norm_{v_2}$ == 0, similarity = 1
16. sim = $dot\_/(norm_{v_1} * norm_{v_2})$
17. **Return** sim

شبه کد (1) گراف فیوژن در نسخه GraphZoom_C

۲-٤ GraphZoom_S

فیوژن در این نسخه تنها از اطلاعات ساختاری برای ایجاد گراف ویژگی استفاده می‌کند برای این منظور کافی است در خط هفتم شبه کد ۱ به جای فراخوانی تابع شباهت کسینوسی از تابع شباهت ساختاری آدامیک آدار (adamic_adar_similarity) استفاده شود. شبه کد ۲، این تابع را نشان می‌دهد.

def adamic_adar_similarity(v_1, v_2, A)
Input: v_1 and v_2 are two nodes and A is the adjacency matrix of G_{topo}
Output: AdamicAdar Similarity between v_1 and v_2

1. $CN = np.nonzero(np.multiply(A[v_1], A[v_2]))[0]$
2. similarity = 0
3 **For** neighbor in CN:
4. $degree = np.sum(A[neighbor])$
5. sim += $\frac{1}{np.\log(degree)}$ **If** degree > 1 **Else** 0
6. **Return** sim

شبه کد (2) معیار مشابهت آدامیک آدار

۳-٤ GraphZoom_SC -۴-۴

می‌توان برای مرحله گراف فیوژن همزمان از اطلاعات محتوایی و ساختاری استفاده کرد: از مشابهت کسینوسی برای محتوا و از مشابهت آدامیک آدار برای مشابهت ساختاری. در GraphZoom_SC یک ترکیب خطی برای ادغام این دو شباهت استفاده شده است. متغیر α مقدار موثر هر یک از شباهتها را کنترل می‌کند. برای اجرای این نسخه، کافی است در خط ۷ شبه کد ۱ تابع Combine_similarities فراخوانی شود که شبه کد ۳ آن را نشان می‌دهد.

1. def Combine_similarities(v_1, v_2, X, G_{topo}, α)
2. S_C= cosine_similarity(v_1, v_2, X)
3. S_AD= adamic_adar_similarity(v_1, v_2 , G_{topo})
4. similarity = α * S_C + (1 - α) * S_AD
5. return similarity

شبه کد (۳) ترکیب معیار های مشابهت کسینوسی و آدامیک آدار

۵ آزمایش

در این بخش سه آزمایش برای مقایسه نسخه‌های پیشنهادی با استفاده از معیارهای Accuracy و زمان اجرا انجام می‌شود. از آنجا که یکی از نسخه‌های مطرح شده (یعنی GraphZoom_C) همان روش اصلی گراف زوم است، عملاً استفاده از اطلاعات محتوایی به جای اطلاعات ساختاری (یعنی GraphZoom_S) و اطلاعات محتولیی در کنار اطلاعات ساختاری (یعنی GraphZoom_SC) در حال آزمایش است.

برای اجرای نسخه‌های پیشنهادی از کدهای موجود در GitHub برای گراف زوم استفاده شده است[۳۱]؛ تنهای کدهای مرتبط با فاز فیوژن تغییر داده شده است. این کدها روی سه مجموعه داده نشان داده شده در جدول (۱) و در محیط گوگل کولب اجرا می‌شود (از توان پردازشی CPU استفاده می‌کنیم). با کمک امبدینگ گره‌های ایجاد شده توسط گراف زوم مسئله پیش‌بینی لینک حل می‌شود و نتایج گزارش شده مربوط به حل این مسئله است.

۵-۲ مسئله پیش‌بینی لینک

در این مسئله، از امبدینگ گره‌ها استفاده می‌شود تا وجود لینک ارتباطی میان گره‌ها پیش‌بینی شود. عملاً باید از روی امبدینگ بدست آمده برای گره‌ها، برای یک جفت گره مشخص امبدینگ ساخته شود. سپس این امبدینگ برای حل یک مسئله رده‌بندی استفاده شود؛ چنانچه خروجی رده‌بند برای یک جفت گره، یک باشد، بین آن جفت گره لینک پیش‌بینی می‌شود. در صورتی که کلاس صفر تشخیص داده شود، بین جفت گره لینک پیش‌بینی نشده است. پس در ادامه دو موضوع باید مشخص شود: نحوه ساخت امبدینگ برای جفت گره و آموزش رده‌بند پیش‌بینی کننده.

برای ساخت امبدینگ برای یک جفت گره، بردارهای امبدینگ آن جفت گره را با عملگر ضرب هادامارد[۳۲] با هم ترکیب می‌کنیم (رابطه ۳ که در آن نشان می‌دهد که درایه i ام ضرب هادامارد دو

بردار h_1 و h_2 حاصلضرب دو درایه iام این دو بردار است). این عملگر را به خاطر کارایی مناسبش در ترکیب امبدینگ گره‌ها برای پیش‌بینی لینک انتخاب کرده‌ایم که در مقاله مرجع [۹] گزارش شده است.

$$\|h_1 \boxdot h_2\|_i = (h_1[i] * h_2[i]) \qquad (۳)$$

ساخت مجموعه داده‌های آموزشی و تست به صورت زیر است: ۵۰ درصد یال‌های موجود در گراف به طور تصادفی از شبکه حذف می‌شوند البته به صورتی که شبکه باقی‌مانده متصل بماند. شبکه باقی مانده برای داده آموزشی لحاظ می‌شود (هر یال موجود در شبکه یک نمونه از کلاس یک یا نمونه مثبت است). برای تولید نمونه‌های منفی (داده‌های با کلاس صفر)، به‌طور تصادفی جفت گره‌هایی را انتخاب می‌کنیم که یالی ندارند. آن ۵۰ درصد یالهای حذف شده برای تست رده‌بند استفاده می‌شود. مقدار Accuracy داده‌های تست در این مقاله گزارش می‌شود.

۵-۳ مجموعه داده‌ها

در این بخش سه مجموعه داده مورد استفاده در آزمایش‌ها معرفی می‌شوند. جدول (۱) اطلاعات کلی مربوط به آنها را نشان می‌دهد. هر سه شبکه انتخاب شده مربوط به استنادات در مقالات علمی هستند. بدین ترتیب هر گره یک مقاله و هر یال ارتباط بین دو مقاله را نشان می‌دهد. مقاله منتسب در گره دارای ویژگی‌هایی است که تعداد آنها در جدول (۱) به ازای هر مجموعه داده مشخص شده است. در ادامه توضیح مختصری از هر شبکه ارایه می‌شود.

جدول ۱: مشخصات مجموعه داده‌های استفاده شده در آزمایش

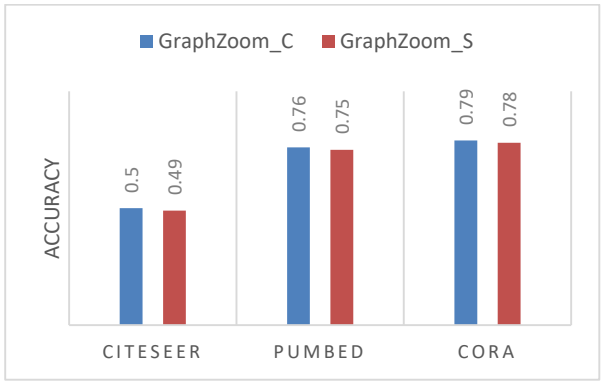
	citeseer	pumbed	Cora
نوع	شبکه استناد	شبکه استناد	شبکه استناد
گره	۳۳۲۷	۱۹۷۱۷	۲۷۰۸
یال	٤٧٣٢	٤٤٣٣٨	٥٤٢٩
ویژگی	۳۷۰۳	۵۰۰	۱٤۳۳

- citeseer: مجموعه ای از مقالات علمی است که در شش کلاس[۳۳] مختلف طبقه‌بندی شده است. هر مقاله با یک بردار باینری نشان داده می شود که وجود یا عدم حضور ۳۷۰۳ کلمه منحصر به فرد از فرهنگ لغت را نشان می دهد. یال بین دو گره جهت‌دار است و ارجاع مقاله مبدأ را به مقاله مقصد نشان می‌دهد.
- pubmed: مجموعه‌ای از مقالات علمی از پایگاه داده PubMed است که منبعی رایگان برای ادبیات زیست پزشکی است. در هر گره یک مقاله است که ویژگی‌هایی دارد که شامل اطلاعاتی مانند عنوان مقاله، نام نویسنده، مجله منتشر کننده، سال انتشار، کلیدواژه‌ها، چکیده و لینک به متن کامل مقاله است. یال‌ها در این مجموعه، نشان‌دهنده ارجاعات یک مقله به مقالات دیگر است. مجموعه داده برای انجام تحلیل‌های مختلف مانند شبکه‌سازی مقالات، پیش‌بینی موضوعات جدید در پزشکی و ارزیابی عملکرد نویسندگان و مجلات پزشکی استفاده می‌شود. مجموعه داده طیف گسترده ای از موضوعات و رشته‌های مرتبط با علوم زیستی مانند پزشکی، زیست شناسی، ژنتیک، فارماکولوژی و غیره را پوشش می دهد.
- Cora: شامل ۲۷۰۸ مقاله علمی است که در هفت کلاس طبقه بندی شده اند. شبکه استناد شامل ٥٤٢٩ یال است. هر مقاله در مجموعه داده با یک بردار باینری ۱٤۳۳ تایی نشان داده شده است که هر درایه باینری آن متناظر با یک کلمه در فرهنگ لغات است؛ مقدار ۱ یعنی وجود کلمه متناظر در متن و مقدار صفر یعنی عدم وجود آن کلمه در متن مقاله.

٥-٤ گراف زوم با ساختار و محتوا

در این آزمایش، با استفاده از معیار Accuracy GraphZoom_C و GraphZoom_S مقایسه شده است. شکل (۲) نتیجه این آزمایش را نشان می‌دهد. همانطور که قبلا گفته

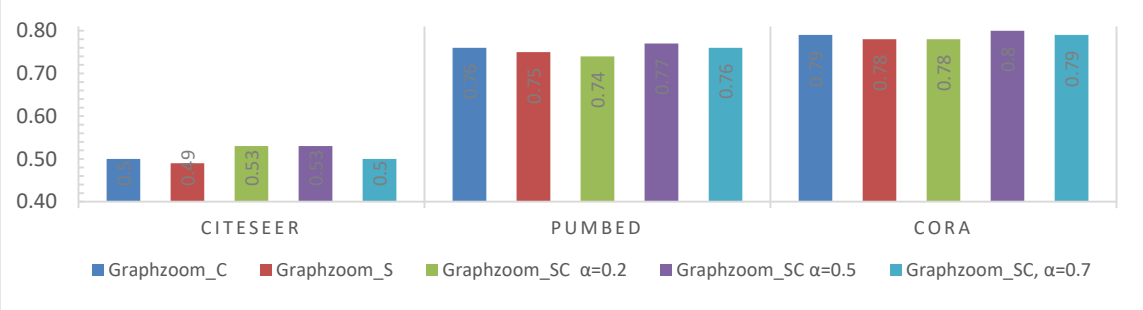
شد، نسخه GraphZoom_C همان نسخه اصلی گراف‌زوم است. این شکل تفاوت زیادی را برای دو نسخه نشان نمی‌دهد.



شکل ۲: مقایسه دو نسخه متفاوت گراف‌زوم؛ محتوایی با ساختاری

۵-۵ ترکیب ساختار و محتوا در گراف زوم

در این آزمایش، سه نسخه پیشنهادی بر اساس معیار Accuracy با هم مقایسه می‌شوند. این آزمایش هم شباهت محتوا و هم شباهت ساختاری را هنگام تجزیه و تحلیل یک گراف در نظر می‌گیرد. برای ایجاد تعادل بین این دو شباهت، ترکیب خطی با متغیر α استفاده می‌شود. در این آزمایش مقدار متغیر از مجموعه $\{0.2, 0.5, 0.7\}$ انتخاب می‌شود. مقدار $\alpha = 0.2$ یعنی در فاز فیوژن، به میزان ۲۰ ٪ از مشابهت کسینوسی و ۸۰٪ از مشابهت آدامیک آدار استفاده شده است. شکل (۳) نتایج این آزمایش را نشان می‌هد.



شکل ۳: مقایسه نسخه‌های پیشنهادی بر اساس Accuracy

جدول 2 : زمان اجرای کامل نسخه‌های پیشنهادی (ثانیه)

	GraphZoom_C	GraphZoom_S	GraphZoom_SC α=0.5	بهبود SC نسب به C
citeseer	۲۲۵.۳۵۱	۲۲۳.۷۹۱	۱۶۹.۴۶۸	24.70%
pumbed	۲۶۳۶.۶۲۶	۲۶۱۱.۴۸۱	۱۷۸۵.۷۰۷	32.20%
cora	۲۱۲.۳۸۴	۲۰۸.۲۳۵	۱۴۱.۷۲۴	33.20%

از شکل (۳) می‌توان به ترکیب مناسب با متغیر $\alpha = 0.5$ اشاره کرد. یعنی ترکیب مساوی ساختار و محتوا Accuracy را در تمام مجموعه‌های افزایش داده است.

۵-۶ مقایسه زمان در نسخه‌های متفاوت

در این آزمایش زمان موردنیاز برای اجرای کامل گراف زوم در سه نسخه پیشنهادی مقایسه می‌شود. معیار Accuracy در ترکیب اطلاعات ساختاری و محتوایی منجر به بهبود جزیی شده است. با مقایسه زمان می‌توان به دید بهتری از نسخه‌های پیشنهادی رسید. جدول (۲) زمان مورد نیاز سه نسخه را نشان می‌دهد.

زمان اجرا الگوریتم گراف زوم با روشهای وزن‌دار کردن پیشنهادی کاهش می‌یابد. در ستون آخر جدول (۲) نشان داده شده است که وقتی از ترکیب ساختار و محتوا استفاده می‌شود، به چه میزان سرعت اجرا افزایش می‌یابد؛ میزان بهبود حداقل ۲۴ درصدی در تمامی مجموعه‌های داده دیده می‌شود.

این نکته نشـان می‌دهد که نسـخه‌های پیشـنهادی موفق شـده‌اند وزن‌های کاراتری به یال‌های گراف انتســـاب دهند. این مقادیر روی کارکرد ســـایر فازهای تاثیر می‌گذارد و ســـرعت را بهبود می‌دهد.

٦ نتیجه‌گیری

در این مقاله، یکی از روشـهای رایج سـلسـله مراتبی در بازنمایی گراف، یعنی گراف زوم، مورد مطالعه قرار گرفت. این الگوریتم چهار فاز اصـــلی دارد که فاز اول (فیوژن) در این مقاله دقیقتر بررسی شد. در الگوریتم اصلی، در فاز اول تنها به محتوای شبکه توجه می‌شـود و در این مقاله اسـتفاده از سـاختار گره‌ها در کنار محتوا مورد توجه بود. بعد از انجام آزمایشـات، مشـخص شـد که توجـه بـه ســـاختـار در کنـار محتوا گراف می‌توانـد بـه بهبود Accuracy در پیش‌بینی لینک منجر شـود. شـاید این میزان بهبود کم باشـــد ولی ســـرعت اجرای الگوریتم گراف زوم با در نظر گرفتن محتوا و ســـاختار در فاز اول، حداقل ٢٤ درصـــد بهتر می‌شود.

مراجع

[1]. M. Defferrard, X. Bresson, and P. Vandergheynst, “Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering,” *Adv Neural Inf Process Syst*, pp. 3844–3852, 2023.

[2]. S. Zhang, H. Tong, J. Xu, and R. Maciejewski, “Graph convolutional networks: a comprehensive review,” *Comput Soc Netw*, vol. 6, no. 1, pp. 1–23, 2019.

[3]. Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, “A Comprehensive Survey on Graph Neural Networks,” *IEEE Trans Neural Netw Learn Syst*, vol. 32, no. 1, pp. 4–24, Jan. 2019.

[4]. C. Deng, Z. Zhao, Y. Wang, Z. Zhang, and Z. Feng, “Graphzoom: A multi-level spectral approach for accurate and scalable graph embedding”, *arXiv preprint arXiv:1910.02370*, 2019.

[5]. G. Valiente, *Algorithms on trees and graphs*, vol. 112, Heidelberg: Springer, 2002.

[6]. T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, New York: springer, 2009.

[7]. M. G. Raeini, “Link Prediction Using Supervised Machine Learning based on Aggregated and Topological Features”, arXiv preprint arXiv:2006.16327, 2020.

[8]. M. Rosvall and C. T. Bergstrom, “Maps of random walks on complex networks reveal community structure,” *Proc Natl Acad Sci U S A*, vol. 105, no. 4, pp. 1118–1123, 2008.

[9]. A. Grover and J. Leskovec, “Node2Vec: Scalable Feature Learning for Networks,” *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, vol. 13, pp. 855–864, 2016.

[10]. H. L. Nguyen, D. T. Vu, and J. J. Jung, “Knowledge graph fusion for smart systems: A Survey,” *Information Fusion*, vol. 61, pp. 56–70, 2020.

[11]. D. C. G. Pedronette, F. M. F. Gonçalves, and I. R. Guilherme, “Unsupervised manifold learning through reciprocal kNN graph and Connected Components for image retrieval tasks,” *Pattern reorganization*, vol. 75, pp. 161–174, 2018.

[12]. H. T. Derek Liu, A. Jacobson, and M. Ovsjanikov, “Spectral Coarsening of Geometric Operators,” *ACM Trans Graph*, vol. 38, no. 4, 2019.

[13]. C. S. Lee, F. Hamon, N. Castelletto, P. S. Vassilevski, and J. A. White, “Nonlinear multigrid based on local spectral coarsening for heterogeneous diffusion problems,” *Comput Methods Appl Mech Eng*, vol. 372, p. 113432, 2020.

[14]. Q. Wang, Z. Mao, B. Wang, and L. Guo, “Knowledge graph embedding: A survey of approaches and applications,” *IEEE Trans Knowl Data Eng*, vol. 29, no. 12, pp. 2724–2743, 2017.

[15]. T. Lv and M. Feng, “A smooth local path planning algorithm based on modified visibility graph”, *Modern Physics Letters B*, vol. 31, no. 19–21, 2017.

[۱] Embedding
[۲] Representation
[۳] Topology
[۴] Graph Representation Learning
[۵] Shallow
[۶] Graph Neural Network
[۷] Graph Convolutional Networks (GCNs)
[۸] این نگاه برای هر دو دسته shallow و شبکه عصبی گرافی قابل اعمال است.
[۹] adjacency matrix
[۱۰] degree matrix
[۱۱] Laplacian matrix
[۱۲] Link prediction
[۱۳] Community detection
[۱۴] recommender systems
[۱۵] Adamic-Adar
[۱۶] K-Nearest Neighbors
[۱۷] Machin Learning
[۱۸] Natural language processing
[۱۹] Cosine Similarity
[۲۰] Random Walk
[۲۱] DeepWalk
[۲۲] Biased Random Walk
[۲۳] breadth-first search
[۲۴] depth-first search
[۲۵] Graph Fusion
[۲۶] Spectral Coarsening
[۲۷] Eigen Vectors
[۲۸] Low-pass graph filtering
[۲۹] Smoothed Vectors
[۳۰] DeepWalk
[31] https://github.com/cornell-zhang/GraphZoom
[32] Hadamard
[33] Agents ،AI ،DB ،IR ،M, HCI

پنجمین کنفرانس ملی انفورماتیک ایران
پژوهشگاه دانشهای بنیادی، پردیس فرمانیه، تهران
۱۳ و ۱۴ دی ماه ۱۴۰۲

بازیابی تصاویر محتوامحور با استفاده از ویژگی‌های بافت استخراج شده از الگوی دودویی محلی دولایه

سید علی حسینی[۱]، امیر حسین عشقی[۲]، صبا محمدی[۳]

[۱] استادیار، دانشکده مهندسی برق و کامپیوتر، دانشگاه بیرجند، بیرجند،
sa.hoseini@birjand.ac.ir

[۲] دانشکده مهندسی برق و کامپیوتر، دانشگاه بیرجند، بیرجند،
amir.eshghi@birjand.ac.ir

[۳] دانشکده مهندسی برق و کامپیوتر، دانشگاه بیرجند، بیرجند،
sabamohammadi.1321@gmail.com

چکیده

بازیابی تصاویر و یافتن الگو های تصاویر با استفاده از هوش مصنوعی امروزه یکی از مهم ترین مسائل در حوزه ی پردازش تصویر است. بازیابی تصاویر ابزار قدرتمندی برای پیدا کردن تصاویر مشابه در یک مجموعه تصاویر بسیار بزرگ می‌باشد. مسئله پیدا کردن الگو ها داخل تصاویر بسیار پیچیده است و راه‌های زیادی برای پیدا کردن الگو ها وجود دارد. یکی از این راه‌ها ساخت و استخراج بافت تصاویر است. برای تولید و استخراج بافت در تصاویر الگوریتم‌های زیادی تاکنون پیشنهاد شده است که مهم‌ترین آنها الگوی دودویی محلی است. در این الگوریتم برای ساخت بافت از همسایه‌های یک پیکسل که در یک فاصله معین قرار گرفته‌اند استفاده می‌شود. در این پژوهش برای ساخت بافت تصاویر از الگوی دودویی محلی دولا یه استفاده شده است که به جای یک فاصله معین از چند فاصله جهت استخراج بافت تصاویر بهره می‌گیرد. این کار سبب می‌شود طیف گسترده‌تری از پیکسل‌ها برای ساخت بافت مشارکت کنند. در روش معرفی شده طول بردار ویژگی بزرگ نمی‌شود و سرعت پردازش‌ها بالاست. همچنین عملکرد این روش در مقایسه با الگوریتم‌هایی که برای ساخت الگوی دودویی محلی پیشنهاد شده‌اند دارای عملکرد مطلوبتری است.

کلمات کلیدی

الگو، بازیابی تصاویر، الگوریتم مبتنی بر محتوا، استخراج ویژگی، الگو دودویی محلی، بافت تصویر، بردار ویژگی .

۱- مقدمه

جستجوی تصاویر محتوا محور (CBIR[*]) راه‌حلی برای مشکل بازیابی تصاویر با استفاده از محتوای تصاویر است. بازیابی تصاویر به معنای یافتن تصاویر از روی محتوای آنها از یک پایگاه داده تعریف می‌شود. نیاز به بازیابی تصاویر مشابه از پایگاه داده‌های بزرگ، امروزه ضرورتی است که برای این مانع باید راه‌حلی سریع و کارا وجود داشته باشد. تصاویر می‌توانند بر اساس اطلاعات فراداده یا محتوای تصویر، پرس‌وجو شوند. برچسب زنی در سامانه‌های سنتی بازیابی تصویر، کاری پیچیده و زمانبر است. هر چه زمان می‌گذرد، بازیابی، پردازش، جستجو، مرور و مدیریت تصاویر دشوارتر می‌شود. برای حل مسئله بازیابی تصویر، تصاویر بر اساس ویژگی‌های سطح بالا یا پایین یا گاهی ترکیبی از هر دو بازیابی می‌شوند. بازیابی تصاویر بر اساس یک پرس‌وجو از پایگاه داده بزرگ انجام می‌شود که در روش های سنتی برای یافتن تصاویر برچسب هایی تعیین می‌شد که امروزه اینکار غیر ممکن است [۱]. امروزه بازیابی تصاویر بر اساس الگوریتم‌های بازیابی شیوه‌ی متفاوتی به خود گرفته است، تصاویر بر اساس محتوایشان که شامل رنگ، بافت، حاشیه، لبه و ... ویژگی های خاصی را به خود اختصاص میدهند، براین اساس الگوریتم های بازیابی سعی می‌کنند این ویژگی‌ها را برای عکس استخراج کنند و تصاویر را براین اساس دسته بندی کنند [۲]. این روش باعث می‌شود جستجوی تصاویر نسبت به روشهای سنتی بازیابیِ اطلاعات که بعضاً بر اساس برچسب‌ها یا نام‌گذاری‌های انسانی انجام می‌شود، موثرتر و دقیق‌تر باشد. به جای وابستگی به کلمات کلیدی یا توصیفات متنی، بازیابی تصویر محتوا محور به صورت مستقیم بر روی ویژگیهای تصویری تمرکز دارد.

استفاده از بازیابی تصویر محتوامحور در زمینه‌های مختلفی از جمله پزشکی (برای تشخیص بیماری‌ها از طریق تصاویر پزشکی)، حیات وحش (شناسایی حیوانات و گیاهان)، مدیریت تصاویر و حتی در فناوری‌های امنیتی، بسیار حائز اهمیت است.

بافت در سال‌های اخیر، به‌عنوان یک ویژگی بصری اساسی برای توصیف تصویر، توجه زیادی به خود جلب کرده است. بافت تصویر قادر است تا جزئیات زیادی را استخراج کند. طبقه‌بندی بافت نقش حیاتی در تجزیه و تحلیل بافت تصویر داشته و به طور گسترده در زمینه‌های مختلفی از جمله کنترل کیفیت

[*] Content-Based Image Retrieval

محصولات صنعتی به‌کار رفته است [۳]. برای استخراج بافت تصاویر به عنوان یک ویژگی، الگوریتم‌های زیادی وجود دارد که یکی از آنها الگوریتم الگوی دودویی محلی است. الگوی دودویی محلی اطلاعات بافت یک تصویر را به صورت محلی استخراج کند. این توصیفگر قادر است بافت های تصویر را در کانال‌های مختلف استخراج کند اما نکته اینجاست که نسخه‌ی ابتدایی این الگوریتم مقدار زیادی از اطلاعات را نادیده میگیرد و نهایتا بخشی از اطلاعات از دست می‌رود [۴]. از این رو الگوریتم‌های بسیار زیادی پیشنهاد شده است تا این الگوریتم اولیه الگوی دودویی محلی را بهبود بخشند.

۲- کارهای مرتبط

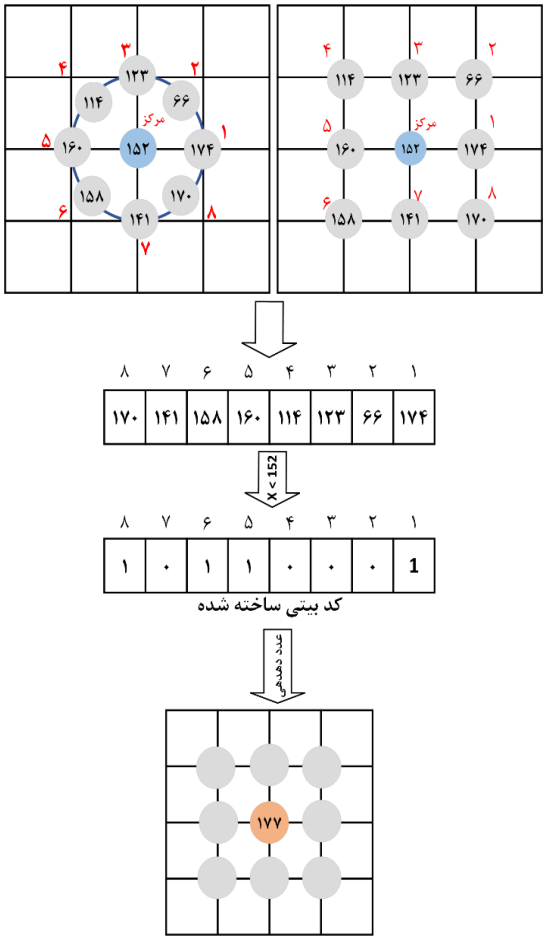
استخراج ویژگی‌های رنگی یک تصویر به طور عموم شامل شناسایی فضای رنگی و کاهش گستره‌ی رنگ می‌شود. توصیفگرهای محبوب رنگ شامل هیستوگرام‌های رنگ، کورلوگرام‌ها، گشتاورهای مرتبط با کانالهای رنگی ، توصیفگرهای رنگی، توصیفگرهای ساختار رنگ و غیره هستند [۵]. یکی دیگر از ویژگیهای سطح پایینِ استخراج شده از تصاویر که معمولا برای توصیف محتوای تصویر از آن استفاده می‌شود، بافت است. بافت تصویر در واقع تابعی از میزان تغییرات مکانی شدت روشنایی پیکسلها است. به عبارت دیگر بافت مشخصه کمّی نواحی یک تصویر می‌باشد. بافت را معمولا با مفاهیمی کیفی نظیر نرم، زمخت، برآمده و ... توصیف می‌کنند که البته برای تحلیل، نیاز به کمّی کردن آن است. یکی از روشهای مشهور برای تعیین بافت تصویر الگوی دودویی محلی است [۶] که نشان دهنده میزان تغییرات شدت روشنایی در مجاورت یک پیکسل می‌باشد. از آنجاییکه این کار با مقایسه شدت روشنایی هر پیکسل با همسایه های آن انجام می‌دهد لذا به لحاظ محاسباتی بسیار ساده و در عین حال بسیار اثربخش است.

۲-۱- الگوی دودویی محلی

الگوی دودویی محلی یکی از متداول‌ترین روش های استخراج بافت تصاویر است که در بسیاری از کارهای حوزه بینایی ماشین برای توصیف بافت تصویر مورد استفاده قرار می‌گیرد. شرح کار این الگوریتم به این صورت است که برای هر پیکسل، تصویر ۸ همسایه آن در یک محیط مربعی یا دایره‌ای در نظر گرفته شده و شدت روشنایی آن‌ها را با مقدار پیکسل مرکزی مقایسه نموده و به ازای مقادیر بزرگتر، یک و به ازای مقادیر کوچکتر، صفر درنظر می‌گیریم. در انتها با کنار هم قرار دادن این صفر و یک‌ها یک کد ۸ بیتی ساخته می‌شود که معادل دهدهی آن یک عدد صحیح در بازه صفر تا ۲۵۵ است. سپس این عدد به عنوان شدت روشنایی پیکسل مرکزی در نظر گرفته می‌شود. فرمول (۱) نحوه محاسبات فوق للذکر را برای یک پیکسل در مختصات (x_c, y_c) نشان می‌دهد.

$$LBP_P(x_C, y_C) = \sum_{P=1}^{P} s(I(x_P, y_P) - I(x_C - y_C)) \times 2^P$$
$$s(I(x_P, y_P) - I(x_C - y_C)) = \begin{cases} 1 & if\ I(x_P, y_P) \geq I(x_C - y_C) \\ 0 & otherwise \end{cases} \quad (۱)$$

در فرمول فوق p نشان دهنده تعداد پیکسل‌های مجاور است که موقعیت آنها بستگی به شعاع همسایگی و نحوه انتخاب همسایه‌ها دارد. در شکل (۱) کیفیت تعیین مقدار الگوی دودویی محلی برای یک پیکسل از تصویر با همسایگی‌های مربعی و دایره‌ای نشان داده شده است. چنانچه این روال برای تمامی پیکسل های تصویر انجام شود در نهایت یک تصویر جدید خواهیم داشت که نمایشی از وضعیت بافت تصویر فراهم می‌کند.



شکل (۱) : نحوه‌ی ساخت بافت در الگوی محلی دودویی اولیه

برای بهبود این الگوریتم اولیه یک راهکار دیگر با همین روش ارائه شده است که بجای درنظر گرفتن همسایه‌ها هر پیکسل در یک محیط مربعی آنها را در یک محیط دایره‌ای نظر گرفته و بدین ترتیب قادر خواهیم بود اطلاعات بهتر و قدرتمندتری از بافت را تولید نماییم [۱۶]. نکته‌ای که راجع به این روش وجود دارد این است که مختصات برخی از نقاطی که بر محیط یک دایره در تصویر قرار دارند دارای مقدار اعشاری است که برای محاسبه مقدار روشنایی تصویر در این نقاط باید از درونیابی دوخطی استفاده نمود.[۱۷]. این الگوریتم مدت زیادی است که درحال گسترش و بهبود عملکرد است. نکته دیگری که در مورد الگوی دودویی محلی باید خاطر نشان ساخت آن است که تجزیه و تحلیل داده‌ها تنها از روی تصاویر خاکستری بدست می‌آید [۱۵] و هیچ توجهی به بافت تصویر در کانال‌های مختلف رنگ نمی‌شود.

یکی از نسخه‌های الگو دودویی محلی که بهبود یافته، الگوی بهینه محلی جهت دار [۷] است که در آن برای یافتن بافت تصویر، تنها از یک جهت استفاده نمی‌شود و درون یک حلقه تمام جهت ها را برای استخراج بافت استفاده می‌کند. الگوی دیگر سه‌گانه محلی [۸] است که در این الگو تنها

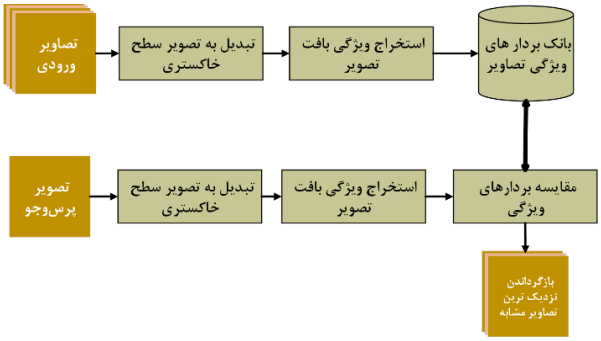
تمرکز برای ساخت یک الگوی بیتی صفر و یک نیست و این مطالعه، الگوی دودویی محلی را به الگوی ســه‌مقداری محلی گســترش می‌دهد که مقادیر تفاضلی بین پیکسل‌های همسایه و پیکسل مرکزی را به عنوان محرک منفی یا مثبت در نظر می‌گیرد. اگر مقدار مطلق تفاضــل بزرگتر باشــد محرک وجود دارد، در غیر این صــورت، هیچ محرکی وجود ندارد. الگوی جهت دار محلی زاویه‌ای (ALDP) [۹] یک نســخه بهبود یافته از الگوریتم الگوی دودویی محلی اســت که در این نسـخه مقدار پیکسـل مرکزی نادیده گرفته می‌شـود. نتایج آزمایشی بر دو مجموعه داده مختلف با استفاده از شش طبقه‌بند مختلف نشان می‌دهد که ALDP به طور قابل توجهی از روش LBP عملکرد بهتری دارد. الگوی ترتیب جهت‌دار محلی (LDOP) [۱۰] روشــی نوآورانه برای ساخت یک توصیفگر محلی با استفاده از همسایگی چندمقیاسی ارائه می‌دهد که با یافتن ترتیب جهت دار محلی بین مقادیر شدت روشنایی در مقیاس‌های مختلف در یک جهت خاص اندازه‌گیری می‌شود. ترتیب جهت‌دار محلی، عامل رابطه چند شــعاعی در یک جهت خاص اســت. الگوی ترتیب جهت‌دار محلی (LDOP) پیشــنهادی برای یک پیکســل خاص از طریق یافتن رابطه بین پیکسل مرکزی و شاخص‌های ترتیب جهتی محلی محاسبه می‌شود. گالشتوار و همکاران [۱۱] یک الگوریتم جدید شاخص‌گذاری تصویر برای بازیابی تصویر مبتنی بر محتوا (CBIR) با اســتفاده از الگوی محلی انرژی‌محور (LEOP) پیشــنهاد داده‌اند. LEOP انرژی و جهت‌های ســطح پیکســل را رمزگذاری می‌کند تا ویژگی‌های مکانی و دقیق تصویر را پیدا کند.

در پژوهشی که اخیرا انجام شده است گونه‌های مختلف محاسبه الگوی دودویی محلی به طور مفصل مورد بحث و بررسی قرار گرفته‌اند [۱۲]. نکته‌ای که لازم اســت در اینجا مورد بررســی قرار گیرد این اســت که تقریبا تمامی روشــهای فوق‌الذکر برای محاســبه الگوی دودویی محلی از تصــویر ســطح خاکستری استفاده نموده‌اند. به عبارت دیگر در این روشها به اطلاعات موجود در کانال‌های رنگ توجه نشده است. برخی از محققان مساله تعیین بافت را از کانالهای رنگ جدا ندیده و به دنبال یافتن مشخصه‌های بافت تصویر در همه کانالهای رنگی بوده‌اند. در همین راســتا و در پژوهشــی که توســط هنگ و همکاران انجام شــده ا ســت LBP های جداگانه برای هر کدام از کانالهای رنگ محاسبه شده و سپس و این ویژگیها با یکدیگر ترکیب شده‌اند [۱۳]. این رویکرد اگرچه عملکرد بهتر بازیابی تصــاویر را به همراه داشــته اســت ولی با معضلی به نام ابعاد بالای بردار ویژگی همراه بوده است. روش عمومی دیگر در این زمینه شــامل اســتخراج ویژگی‌های رنگ و بافت از کانال‌های رنگی تشــکیل‌دهنده تصــویر و ســپس ادغام آنها در یک بردار ویژگی اســت. بدیهی است در این حالت ابعاد بردار ویژگی سه برابر حالتی است که تنها از یک کانال رنگی استفاده شده است [۱۴].

در مقابل برخی از محققین به طور کلی ویژگی‌های رنگ و بافت را به صــورت جداگانه اســتخراج نموده و ســپس همه آنها را در یک بردار ویژگی تجمیع نموده‌اند. این روشــها به طور قابل ملاحظه‌ای ابعاد بردار ویژگی را کاهش می‌دهد.

۳- روش پیشنهادی

الگوریتم پیشنهادی برای بازیابی تصاویر محتوا محور صرفا از ویژگی‌های استخراج شده بافت استفاده نموده است. چارچوب کلی الگوریتم پیشنهادی در شکل (۲) نشان داده شده است. همانگونه که دیده می‌شود در روش پیشنهادی نیز مانند غالب روشهای بازیابی تصاویر محتوامحور از ویژگی‌های بافت استفاده شده است. همانگونه که پیشتر اشاره شد LBP یک روش محبوب برای کمی کردن کیفیت بافت در تصــاویر می‌باشــد. در روش ارائه شــده برای اســتخراج ویژگی‌های بافت از تصویر سطح خاکستری استفاده می‌شود لذا نیاز است به عنوان پیش پردازش ابتدا تصــویر رنگی را به تصــویر ســطح خاکســتری تبدیل نماییم. در این پژوهش گونه جدیدی از روش LBP برای تولید بافت ارائه شده است.



شکل (۲) : نحوه کارکرد الگوریتم بازیابی تصویر پیشنهادی

در روش پیشــنهادی برای تولید بردار ویژگی بر مبنای ســاختار بافت، از یک همسایگی دو لایه استفاده شده است. به عبارت دیگر بافت در هر نقطه به گونه‌ای محاســبه می‌شــود که از پیکســل‌های دورتر از پیکســل مرکزی نیز استفاده می شود. در ادامه به جزئیات روش پیشنهادی پرداخته شده است.

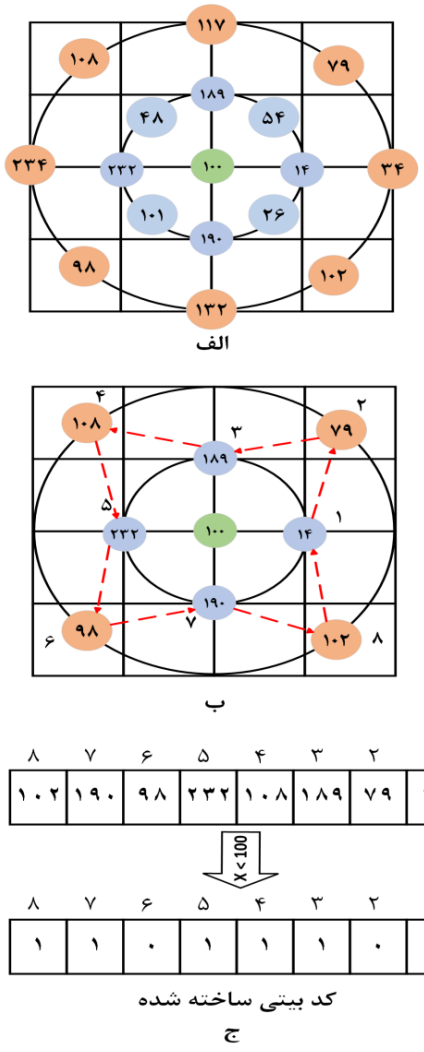
۳-۱- الگوی دودویی محلی دو لایه (BLBP†)

در این روش برای محاســبه مقدار کمی بافت در هر پیکســل تصــویر از مقادیر شدت روشنایی پیکسل‌ها در همسایگی لایه اول و دوم استفاده شده است. مطابق آنچه قبلا گفته شد همسایگی‌ها می توانند به صورت مربعی و یا دایره‌ای باشند. در الگوریتم پیشنهادی همسایگی‌ها به صورت دایره‌ای در نظر گرفته شده‌اند که نتایج بهتری به همراه دارد. در شکل (۳) نحوه محاسبه مقدار بافت برای پیکسل مرکزی که به رنگ سبز مشخص شده است، نشان داده شده است. همانگونه که در شکل (۳-الف) دیده می‌شود مقادیر شدت روشنایی در ۸ پیکسل مجاور در دو همسایگی به شعاع‌های ۱ و ۲ پیکسل نشان داده شــده اســت. نکته مهمی که وجود دارد این اســت که اگر از همه ۱۶ پیکســل موجود بر روی دو شعاع همســایگی استفاده کنیم در این صورت مقادیر بافت برای هر کدام از پیکســل‌ها دارای ۱۶ بیت خواهد که به نوبه خود دارای 2^{16} حالت مختلف خواهد بود. با توجه به اینکه بردار ویژگی متناظر با بافت در هر تصویر در واقع همان هیستوگرام تصویر الگوی دودویی محلی است، این مساله باعث خواهد شــد که طول بردار ویژگی هر تصــویر دارای ۶۵۵۳۶ = 2^{16} مولفه باشــد. اگرچه بردار ویژگی با این اندازه احتمالا قدرت متمایزکنندگی بیشــتری

† Bilayer Local Binary Pattern

دارد ولی برای مقایسه بردار ویژگی‌ها نیاز به هزینه پردازشی هنگفتی دارد. چنانچه تعداد تصاویر موجود در پایگاه داده تصاویر خیلی زیاد باشد عملا مقایسه بردارهای ویژگی عملی بسیار زمانبر خواهد بود. در روش پیشنهادی ابتکاری به‌کار گرفته شده است که علاوه بر اینکه از پیکسلهای دو لایه همسایگی برای تولید ویژگی بافت استفاده می‌کند، طول بردار ویژگی تولید شده نیز برابر ۲۵۶ = 2^8 مولفه خواهد بود.

رویکرد پیشنهاد شده در پژوهش حاضر، مسیر مشابهی با الگوریتم اولیه الگوی دودویی محلی را طی می‌کند، با این تفاوت که در انتخاب همسایگان، شعاع‌ها به صورت متناوب بین هر دو شعاع پیشنهادی جابجا می‌شوند (شکل ۳–ب). این موضوع باعث می‌شود حجم محاسبات کمتری داشته باشیم زیرا تعداد همسایگان به ۸ عدد کاهش خواهد یافت و بردار ویژگی هم دارای 2^8 درایه خواهد بود (شکل ۳–ج). همچنین باعث می‌شود دید گسترده‌تری برای ساخت بافت داشته باشیم و کیفیت تصویرِ بافتِ ساخته شده نیز بهتر خواهد بود.



شکل (۳) : نحوه ی کار کرد الگوریتم SLBP
الف) چینش اولیه‌ی همسایگان پیکسل مرکزی در دو شعاع
ب)نحوه ترتیب انتخاب همسایگان
ج)نحوه ساخت کد بیتی و محاسبه مقدار بافت پیکسل مرکزی

۴- آزمایش و نتایج

برای ارزیابی الگوریتم پیشنهادی آزمایش‌هایی انجام شده است که در ادامه به جزئیات آنها پرداخته می‌شود. قابل ذکر است که روش پیشنهادی با استفاده از زبان برنامه‌نویسی پایتون و بر روی یک سیستم کامپیوتری با پردازنده INTEL CORE I7 و ۱۶ GB حافظه RAM اجرا شده است. همچنین پیاده‌سازی‌ها در محیط ویندوز ۱۱ انجام شده‌اند.

برای ارزیابی روش پیشنهادی از مجموعه داده ولنگ (Wang) [۱۸] استفاده شده است. این مجموعه از ۱۰۰۰ تصویر رنگی تشکیل شده است که اندازه تصاویر آن ۲۵۶×۳۸۴ یا ۳۸۴ × ۲۵۶ است. تصاویر در ۱۰ دسته قرار دارند و هر دسته شامل ۱۰۰ تصویر است. این دسته‌ها شامل تصاویری با محتوای مردم آفریقایی، ساحل، ساختمان، اتوبوس، دایناسور، فیل، گل، اسب، منظره و غذا می‌شوند.

نکته مهم دیگر این است که در الگوریتم ارائه شده از همه تصاویر موجود در مجموعه داده به عنوان تصویر پرس و جو استفاده شده است و همچون کلیه الگوریتم‌های بازیابی تصویر محتوا محور در این پژوهش نیز برای سنجش عملکرد الگوریتم پیشنهادی از معیار دقت (precision) استفاده شده است. بر این اساس، مقدار دقت از رابطه زیر محاسبه می‌شود.

$$P(N) = \frac{I_N}{N} \qquad (۲)$$

که در فرمول فوق P(N) مقدار دقت است و I_N نیز تعداد تصاویر بازیابی شده درست (تعداد تصاویری که به درستی بازیابی شده‌اند) است. N تعداد کل تصاویر بازیابی شده است. به عبارت دیگر برای هر تصویر پرس و جو N تصویر بازیابی می‌شود که تعداد I_N مورد متعلق به کلاسِ تصویر پرس و جو هستند و سپس با کمک فرمول (۲) مقدار دقت برای تصویر پرس و جوی فعلی محاسبه می‌شود. این فرایند برای همه تصاویر مجموعه داده تکرار شده و سپس میانگین مقادیر دقت آنها بدست می‌آید. در آزمایشهای انجام شده مقدار N برابر ۱۰ در نظر گرفته شده است و بر این اساس در جدول (۱) مقدار دقت الگوریتم پیشنهادی به تفکیک کلاس‌ها و همچنین نوع فاصله مورد استفاده برای مقایسه بردارهای ویژگی درج شده است.

جدول(۱) دقت الگوریتم پیشنهادی بر روی مجموعه داده wang و به تفکیک دسته و نوع فاصله انتخابی

	Extended Canberra	Canberra	Square chord	Euclidean
مردم آفریقایی	**۶۹/۹**	۶۶/۷	۶۵/۶	۶۳/۴۴
ساحل	**۵۷/۳**	۵۳/۸	۵۱/۸	۴۷/۴۱
ساختمان	۵۸	**۵۸/۳**	۵۵/۸۶	۴۹/۰۹
اتوبوس	**۹۶/۵**	۹۶/۳	۹۶/۲	۹۴
دایناسور	**۹۸/۶**	۹۷/۸	۹۷/۹	۴.۹۷
فیل	**۵۴/۶**	۵۳/۹	۵۰/۱	۴۴/۹
گل	**۹۱/۸**	۹۰/۷۹	۸۸/۲	۷۹/۴
اسب	**۸۴/۳**	۸۳/۳	۷۹/۸	۷۰
منظره	۳۴/۸	**۳۴/۹**	۳۱/۶	۲۶/۷
غذا	**۵۵/۵**	۵۴/۷۹	۵۴/۰۹	۴۲/۴۸

در ادامه به منظور بررسی کیفیت عملکرد روش پیشنهادی، آنرا با تعدادی از روشهای مشابه که از اطلاعات بافت برای بازیابی تصاویر استفاده نموده‌اند

مقایسه کرده ایم. بر این اساس، تلاش شده روش‌های ارائه شده توسط سایر پژوهشگران پیاده‌سازی شده و عملکرد آنها بر روی مجموعه داده wang محاسبه شود. در این راستا الگوریتمهای LBPH [۱۹]، MdLBP [۲۰]، LBP [۱۶] و QLBP [۲۱] پیاده‌سازی شده و نتایج آنها با روش پیشنهادی مقایسه شود. در جدول (۲) نتایج مقایسه بر روی مجموعه داده wang و برای حالت N=10 تصویر بازیابی شده نمایش داده شده است. در این مقایسه‌ها برای محاسبه میزان شباهت بردارهای ویژگی از فاصله کانبرا توسعه یافته استفاده شده است.

جدول(۲) مقایسه دقت روش پیشنهادی با سایر روشها

نام روش	اندازه بردار ویژگی	دقت
LBP classic[16]	۲۵۶	۶۷/۰۴
LBPH[19]	۲۵۶	۶۹/۲
QLBP[21]	۲۵۶	۶۷/۵۵
MDLBP[20]	۲۵۶	۶۹/۳
BLBP	۲۵۶	**۷۰/۱۱**

۵- نتیجه گیری

در این پژوهش یک چارچوب مبتنی بر الگوی دودویی محلی برای مساله بازیابی تصاویر محتوا محور ارائه شد. برخلاف الگوریتم‌های اولیهِ محاسبه الگوهای دودویی محلی، در الگوریتم ارائه شده از یک همسایگی وسیعتر استفاده شده است. راهکار پیشنهادی برای اجتناب از بزرگ شدن بردار ویژگی متناظر با هر تصویر، مقایسه میان پیکسل مرکزی را با پیکسلهای قرار گرفته بر روی دو همسایگی با شعاعهای متفاوت به صورت یکی درمیان انجام داده است. با وجود آنکه فقط از ویژگی‌های بافت استفاده شده است اما در مقایسه با بسیاری از روش‌های مشابه عملکرد بهتری از خود نشان داده است. در پژوهشهای آینده سعی بر آن است با ترکیب سایر ویژگیهای استخراج شده از تصویر مانند ویژگی‌های رنگ و شکل بتوان عملکرد روش پیشنهادی را ارتقا داد.

مراجع

[1] G. S. Vieira, A. U. Fonseca, and F. Soares, "*CBIR-ANR: A content-based image retrieval with accuracy noise reduction,*" Software Impacts, vol. 15, p. 100486, 2023.

[2] Y. Rui, T. S. Huang, and S.-F. Chang, "*Image retrieval: Current techniques, promising directions, and open issues,*" Journal of visual communication and image representation, vol. 10, no. 1, pp. 39-62, 1999.

[3] S. Lan, X. Liao, H. Fan, S. Hu, and Z. Pan, "*A multi-channel framework based Local Binary Pattern with two novel local feature descriptors for texture classification,*" Digital Signal Processing, vol. 140, p. 104124, 2023.

[4] A. Khan, A. Javed, M. T. Mahmood, M. H. A. Khan, and I. H .Lee, "*Directional magnitude local hexadecimal patterns: A novel texture feature descriptor for content-based image retrieval,*" IEEE Access, vol. 9, pp. 135608-135629, 2021.

[5] E. R. Vimina and P. Jacob, "*Computational frameworks for efficiency enhancement of content-based image retrieval systems,*" 2018.

[6] Ojala, M. Pietikainen, and T. Maenpaa, "*Multiresolution gray-scale and rotation invariant texture classification with local binary patterns,*" IEEE Transactions on pattern analysis and machine intelligence, vol. 24, no. 7, pp. 971-987, 2002.

[7] T. Chakraborti, B. McCane, S. Mills, and U. Pal, "*LOOP descriptor: local optimal-oriented pattern,*" IEEE Signal Processing Letters, vol. 25, no. 5, pp. 635-639, 2018.

[8] X.-H. Han, G. Xu, and Y.-W. Chen *"Robust local ternary patterns for texture categorization,*" 6th International Conference on Biomedical Engineering and Informatics, pp. 846-850., 2013.

[9] A. M. M. Shabat and J. R. Tapamo, "*Angled local directional pattern for texture analysis with an application to facial expression recognition,*" IET Computer Vision, vol. 12, no. 5, pp. 603-608, 2018.

[10] S. R. Dubey and S. Mukherjee, "*Ldop: local directional order pattern for robust face retrieval,*" Multimedia Tools and Applications, vol. 79, pp. 6363-6382, 2020.

[11] G. M. Galshetwar, L. M. Waghmare, A. B. Gonde, and S. Murala, "*Local energy-oriented pattern for image indexing and retrieval,*" Journal of Visual Communication and Image Representation, vol. 64, p. 102615, 2019.

[12] R. Arya and E. R. Vimina, "*An evaluation of local binary descriptors for facial emotion classification,*" Innovations in Computer Science and Engineering: Proceedings of 7th ICICSE, pp. 195-205, 2020.

[13] G. M. Galshetwar, L. M. Waghmare, A. B. Gonde, and S. Murala, "*Edgy salient local binary patterns in inter-plane relationship for image retrieval in diabetic retinopathy,*" Procedia computer science, vol. 115, pp. 440-447, 2017.

[14] S. Singh and S. Batra, "*An efficient bi-layer content-based image retrieval system,*" Multimedia Tools and Applications, vol. 79, no. 25-26, pp. 17731-17759, 2020.

[15] L. Wang and D.-C. He, "*Texture classification using texture spectrum,*" Pattern recognition, vol. 23, no. 8, pp. 905-910, 1990.

[16] D.-C. He and L. Wang, "*Texture unit, texture spectrum, and texture analysis,*" IEEE transactions on Geoscience and Remote Sensing, vol. 28, no. 4, pp. 509-512, 1990.

[17] Y. Sa, "*Improved bilinear interpolation method for image fast processing,*" 7th IEEE International Conference on Intelligent Computation Technology and Automation, 2014.

[18] J. Z. Wang, L. Jia, and G. Wiederhold, "*SIMPLIcity: semantics-sensitive integrated matching for picture libraries,*" IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 9, pp. 947-963, 2001.

[19] C. Singh, E. Walia, and K. P. Kaur, "*Color texture description with novel local binary patterns for effective image retrieval,*" Pattern Recognition, vol. 76, pp. 50-68, 2018.

[20] S. R. Dubey, S. K. Singh, and R. K. Singh, "*Multichannel Decoded Local Binary Patterns for Content-Based Image Retrieval,*" IEEE Transactions on Image Processing, vol. 25, no. 9, pp. 4018-4032, 2016.

[21] R. Lan, Y. Zhou, and Y. Y. Tang, "*Quaternionic local ranking binary pattern: a local descriptor of color images,*" IEEE Transactions on Image Processing, vol. 25, no. 2, pp. 566-579, 2015.

پنجمین کنفرانس ملی انفورماتیک ایران
پژوهشگاه دانشهای بنیادی، پردیس فرمانیه، تهران
۱۳ و ۱۴ دی ماه ۱۴۰۲

الگوریتم ژنتیک چند هدفه مرتب سازی نامغلوب مبتنی بر خوشه‌بندی فازی

پژمان غلام نژاد[۱]، امیر مهدی سازدار[۲]، عبدالله غفاری[۳]

[۱] دانشکده مهندسی رایانه و سایبر، دانشگاه علوم و فنون هوایی شهید ستاری، تهران
pezhman.gholamnezhad@gmail.com

[۲] دانشکده مهندسی رایانه و سایبر، دانشگاه علوم و فنون هوایی شهید ستاری، تهران
sazdar@gmail.com

[۳] دانشکده مهندسی رایانه و سایبر، دانشگاه علوم و فنون هوایی شهید ستاری، تهران

Abdollah.ghaffari@ut.ac.ir

چکیده

الگوریتم ژنتیک چند هدفه مرتب سازی نامغلوب[1] یکی از شاخص‌ترین و پرکاربردترین روش‌های چند هدفه تکاملی در زمینه بهینه‌سازی می‌باشد. این الگوریتم بارها توسط افراد مختلف، برای ایجاد الگوریتم‌های بهینه‌سازی چندهدفه جدیدتر، مورد تغییرات جدید قرار گرفته است که عمده این تغییرات مبتنی بر قوانین ثابت اکتشافی مانند تقاطع و جهش بوده است. در این الگوریتم، در ابتدا رتبه‌بندی افراد نامغلوب، بر اساس رتبه و فاصله ازدحام انجام می‌شود و عملگرهای انتخاب، تقاطع و جهش برای تولید فرزندان، اجرا می‌شوند و سپس ترکیب جمعیت والدین و فرزندان برای شکل‌گیری جمعیت جدید انجام می‌شود و در انتها، انتخاب جمعیت جدید، بر اساس رتبه‌بندی و فاصله ازدحام صورت می‌پذیرد. در روش پیشنهادی، برای انتخاب جمعیت جدید، محاسبه فاصله ازدحام، بر اساس الگوریتم فازی مبتنی بر خوشه-بندی صورت می‌پذیرد که منجر به دقت بالاتر در انتخاب افراد دارای رتبه بالاتر، در جمعیت جدید، می‌گردد. نتایج روش پیشنهادی در سکوی ای اِم اُ[2]، بر روی توابع تست، مورد ارزیابی قرار گرفته است و با روش-های مشابه مقایسه شده است. نتایج نشان می‌دهد که با تعداد تکرار کمتر، نتایج بهتری حاصل می‌شود.

کلمات کلیدی

الگوریتم ژنتیک چند هدفه مرتب‌سازی نامغلوب، خوشه بندی فازی، بهینه‌سازی چند هدفه تکاملی.

۱- مقدمه

ساختار پایه بیشتر الگوریتم‌های تکاملی چند هدفه سنتی مبتنی بر قوانین ثابت اکتشافی مانند تقاطع و جهش است که دارای فرآیندهای قوی اکتشافی در فضای تصمیم[3] می‌باشند. از سال ۲۰۰۰ تا کنون، الگوریتم‌های تکاملی چند هدفه بسیاری ارائه شده‌اند که در سه گروه دسته‌بندی شده‌اند(Ma et al., 2021): روش‌های مبتنی بر وزن‌دهی مجتمع[4]، روش‌های مبتنی بر غلبگی[5]، و نگرش مبتنی بر کارایی شاخص[6]. در روش‌های وزن‌دهی مجتمع، یک مساله بهینه‌سازی چند هدفه به یک تعداد از مسائل بهینه‌سازی تک هدفه، با استفاده از یک تعداد ترکیبات وزنی، که به‌صورت تصادفی ایجاد می‌شوند، تجزیه می-شود.

در الگوریتم‌های تکاملی چند هدفه مبتنی بر غلبگی، از نخبه‌گرایی استفاده شده است که همگرایی در این الگوریتم‌ها، به‌صورت قابل ملاحظه‌ای افزایش یافته است. در این گروه از الگوریتم‌ها با افزایش تعداد اهداف، همگرایی به‌صورت قابل ملاحظه‌ای کاهش می‌یابد که عمدتاً به دلیل کاهش فشار انتخاب است.

یکی دیگر از ایده‌ها، اختصاص یک برازش به افراد یک الگوریتم بهینه-سازی چند هدفه مبتنی بر شاخص کارایی است. در این الگوریتم‌ها، آَبر حجم[7]، می‌تواند هم دقت و هم تنوع مجموعه راه‌حل‌های نامغلوب را محاسبه نماید. از مهم‌ترین چالش‌های این گروه از الگوریتم‌ها،

[1] NSGA-II (Non-Dominated Sort Genetic Algorithm)
[2] Plat EMO

[3] Decision Space
[4] Weighted Aggregation based method
[5] Dominance based method
[6] Indicator based method
[7] Hypervolume

پیچیدگی محاسباتی برای محاسبه شاخص کارایی، در زمانی که زمانی که تعداد اهداف زیاد است، می‌باشد.

بیشتر الگوریتم‌های فوق، بر بسط و توسعه یک برآورد برازش مؤثر و یا استراتژی انتخاب، تمرکز نموده‌اند که منطبق بر یک الگوریتم تکاملی تک هدفه، است و توجه خاصی به طراحی استراتژی بازتولید نشده است که در توزیع راه‌حل‌های بهینه پرتو، تاثیر زیادی دارند.

مهم‌ترین موضوعی که در الگوریتم‌های تکاملی باعث کند شدن روند یافتن پاسخ بهینه می‌شود، تکرارهای متوالی و به نوعی حلقهٔ اصلی تولید نسل است.

۲- مفاهیم و تعاریف پایه

الگوریتم‌های تکاملی به‌عنوان ابزاری مؤثر در مسائل بهینه‌سازی چند هدفه ارائه شده‌اند که به‌طورکلی شامل دو یا سه هدف متناقض است. مسائل بهینه‌سازی چند هدفه، مربوط به مسایلی با M هدفِ متضاد به طور هم‌زمان است که در آن مقدار M، برابر با دو یا سه است(Li et al., 2015) که توسط معادله (۱) توصیف شده است.

$$\begin{cases} f(x) = [f_1(x), \ldots, f_M(x)] \\ s.t. \quad x \in X \end{cases} \qquad (۱)$$

که $X \in \Re^n$، فضای تصمیم است و $f : X \to \Re^M$ فضای هدف[1] است و فرض بر این است که f، یک مساله کمینه‌سازی است.

۲-۱- خوشه‌بندی فازی c-میانگین[2]

در مسائل شناسایی الگو مفهوم خوشه، مجموعه‌ای از داده‌ها است که به علت شباهت زیادی که به هم دارند در یک گروه قرارگرفته‌اند. در خوشه‌بندی، داده‌ها به‌صورت بدون ناظر به خوشه‌هایی تقسیم می‌شوند، بطوری که شباهت داده‌های درون هر خوشه حداکثر و شباهت بین داده‌های درون خوشه‌های مختلف حداقل گردد. یکی از رایج‌ترین روش‌های خوشه‌بندی، خوشه‌بندی فازی c-میانگین است(Pantula et al., 2020).

این الگوریتم از یک روش ساده برای خوشه‌بندی مجموعه داده، در یک تعداد خوشه از پیش مشخص‌شده، استفاده می‌کند. ایده اصلی در نظر گرفتن k مرکز برای هر یک از خوشه‌ها است.

بنابراین مراکز، در فاصله هر چه بیشتر از یکدیگر قرار داده می‌شوند. سپس، هر الگو به نزدیک‌ترین مرکز، اختصاص داده می‌شود. سپس یک گروه‌بندی اولیه انجام می‌شود. سپس k مرکز جدید برای خوشه‌های مرحله قبل ایجاد می‌شود و سپس، مجدداً داده‌ها را به مراکز مناسب تخصیص داده می‌شود. این مراحل را آن‌قدر تکرار می‌شود تا k مرکز، جابه‌جا نشوند.

۲-۲- خوشه‌بندی گاستافسون-کِسِل[3]

در واقع، این الگوریتم توسعه یافته‌ی الگوریتم c-میانگین فازی می‌باشد(Škrjanc et al., 2019) که در آن، برای پیداکردن خوشه‌هایی با اشکال هندسی متفاوت، بر روی مجموعه داده‌ها، فاصله نرمال تطبیقی[4]، به کار گرفته‌شده است(Babuka et al., 2002). الگوریتم میانگین فازی، جستجوی خوشه‌ها را به‌صورت کرَوی انجام می‌دهد، اما این روش خوشه‌ها را به‌صورت بیضی‌شکل پیدا می‌کند. در این روش هر خوشه به‌وسیله مرکز و ماتریس کوواریانس، مشخص می‌شود. تابع هدف در این الگوریتم به‌صورت رابطه زیر، تعریف می‌شود:

$$J(x;U,V,A) = \sum_{i=1}^{c}\sum_{k=1}^{N} (\mu_{ik})^m \, D^2_{ikA_i} \qquad (۴)$$

که در آن:

$$D^2_{ikA_i} = (x_k - v_i)^T \, A_i \, (x_k - v_i) \qquad (۵)$$

می‌باشد. این الگوریتم از فاصله نرمال ماهال[5]، استفاده می‌کند. در این دو رابطه، μ_{ik} درجه عضویت هر فرد به هر خوشه می‌باشد. $D^2_{ikA_i}$ فاصله نرمال ماهال بین نقطه x_k و خوشه k ، می‌باشد. A ماتریس کوواریانس خوشه‌ها می‌باشد. V مراکز خوشه‌ها است.

۲-۳- الگوریتم ژنتیک مرتب‌سازی نامغلوب[6]

در این الگوریتم (Deb et al., 2002)، ابتدا جمعیت فرزندان، با استفاده از جمعیت والدین ساخته می‌شود و به جای پیدا کردن جواب‌های نامغلوب از فرزندان، ابتدا دو جمعیت والدین و فرزندان با یکدیگر ترکیب می‌شوند و سپس از یک مرتب‌سازی نامغلوب، برای دسته‌بندی تمام جمعیت استفاده می‌شود.

در این شیوه، یک مقایسه عمومی بر روی فرزندان و والدین انجام می‌گیرد و پس از ایجاد صف‌های نامغلوب به ترتیب اولویت (اولویت صف‌ها نسبت به هم)، جمعیت بعدی یکی‌یکی از این صف‌ها پر می‌شود. پر کردن جمعیت جدید با بهترین صف نامغلوب شروع می‌شود و سپس به ترتیب با دومین صف نامغلوب و همین‌طور سومین و الی آخر، تا زمانی که جمعیت جدید پر شود، ادامه پیدا می‌کند. از آن‌جا که اندازه جمعیت ما دو برابر است، تمام اعضا در جمعیت جدید قرار نمی‌گیرند و بایستی جمعیت باقیمانده را حذف کنیم. در مورد جواب‌هایی که در صف آخر با استفاده از عملگر نخبه‌گرایی از بین می‌رود، باید مهارت بیشتری به‌کاربرده و جواب‌هایی را که در ناحیه ازدحام کمتری قرار دارند، حفظ کرد. در واقع برای رعایت اصل چگالی در بین جواب‌ها، جواب‌هایی که در ناحیه ازدحام کوچک‌تری هستند، برای پر کردن جمعیت جدید در اولویت قرار دارند.

چارچوب کلی این الگوریتم در شکل (۱) بیان‌شده است:

---

[1] Objective space
[2] Fuzzy C-mean Clustering (FCM)
[3] Gustafson - Kessel
[4] adaptive distance norm
[5] Mahalanibis distance norm
[6] Non-dominated Sorting Genetic Algorithm (NSGA-II)

شکل (۱). چارچوب کلی الگوریتم ژنتیک مرتب‌سازی نامغلوب

۳- روش پیشنهادی

در این مقاله، یک الگوریتم بهینه‌سازی چند هدفه تکاملیِ مرتب‌سازی نامغلوب با استفاده از خوشه‌بندی فازی پیشنهاد شده است.
چارچوب کلی الگوریتم پیشنهادی در شکل (۲) بیان‌شده است:

شکل (۲). چارچوب کلی الگوریتم پیشنهادی

٤- ارزیابی کارآیی

طی دهه‌های اخیر، تعداد زیادی الگوریتم تکاملی برای حل مسائل بهینه‌سازی چند هدفه ایجادشده است. با این وجود، عدم وجود یک بستر نرم‌افزاری به‌روز و جامع برای پژوهشگران به منظور ارزیابی صحیح الگوریتم‌های موجود، برای حل مسائل می‌تواند به‌عنوان یک چالش مطرح باشد. وقتی کد منبع بسیاری از الگوریتم‌های پیشنهادی در دسترس عموم قرار نگرفته باشد، به‌منظور عدم مواجه با چنین چالشی، در این تحقیق، از یک سَکّو[1] در بستر نرم‌افزار متلب[2] برای بهینه‌سازی چند هدفه تکاملی، به نام پلَت ای‌ام آ ، استفاده‌شده است که شامل بیش از ۵۰ الگوریتم تکاملی چند هدفه و بیش از ۱۰۰ مسئله آزمون چند هدفه، همراه با چندین عملکرد پرکاربرد است. این سکو کاملاً متن‌باز است، به‌گونه‌ای که کاربران می‌توانند بر اساس آن، الگوریتم‌های جدیدی بسازند.

کد منبع پلَت ای‌ام آ، در آدرس زیر قابل‌دسترسی است:
http://bimk.ahu.edu.cn/index.phps=/Index/Softwar/index.html.

این نرم‌افزار مبنای حل دقیق برای مدل پیشنهادی این پژوهش است و هدف اعتبارسنجی مدل است.

در این پژوهش، در دامنه وسیعی، آزمایشاتی به‌منظور نشان دادن عملکرد مدل مفهومی پیشنهادی انجام‌شده است و روش پیشنهادی با یک سری از الگوریتم‌ها، مقایسه و ارزیابی‌شده‌اند. این آزمایش‌ها بر روی نمونه تست‌های مشخص انجام‌شده است. بدین منظور در ادامه این بخش، ابتدا معیارهای ارزیابی و شاخص‌های مورد مطالعه معرفی‌شده‌اند. در بخش دوم، تنظیماتِ پارامترها در آزمایش‌ها را نشان می‌دهد. در بخش سوم، الگوریتم‌های مقایسه شده را نشان می‌دهد. بخش چهارم به معرفی توابع ارزیابی می‌پردازد. در بخشِ یافته‌های تحقیق، نتایج آزمایش‌ها نشان داده می‌شود و در بخش بحث و نتیجه‌گیری، بررسی الگوریتم‌های مقایسه و ارزیابی‌شده را نشان می‌دهد.

٤-١- معیارهای ارزیابی و شاخص‌های مورد مطالعه

یک روش ساده برای ارزیابی کیفیت مجموعه‌ی راه‌حل‌ها، شاخص‌های کیفیت[3] است. به‌طورکلی، شاخص‌های کیفیت به شش گروه زیر دسته‌بندی می‌شوند(Li & Yao): ۱- شاخص کیفیت برای همگرایی[4] ۲- شاخص کیفیت برای انتشار[5] ۳- شاخص کیفیت برای یکنواختی[6] ۴- شاخص قدرتمندی[7] ۵- شاخص کیفیت برای انتشار و یکنواختی ۶- شاخص کیفیت برای چهار بخش اول.

دو گروه همگرایی در شاخص‌های کیفیت وجود دارد:
۱-ارزیابی رابطه‌ی غلبگی پرتو[8] بین راه‌حل‌ها یا مجموعه‌ها.
۲- ارزیابی فاصله‌ی یک مجموعه راه‌حل از پرتو نما.

شاخص فاصله نسلی[9]، میانگین مربع فاصله اقلیدسی مجموعه‌ی راه‌حل‌ها، نسبت به نزدیک‌ترین نقطه در پرتو نما را اندازه‌گیری می‌کند(Van Veldhuizen & Lamont, 1998). کیفیت انتشار[10]، مربوط به پوشش ناحیه یک راه حل است. تنوع

[1] Platform
[2] Matlab

[3] Quality Indicator
[4] Convergence
[5] Spread
[6] Uniformity
[7] Cardinality
[8] Pareto dominance
[9] Generational Distance (GD)
[10] Spread quality

خالص[1]، عدم شباهت هر راه حل به بقیه‌ی راه‌حل‌ها را در یک مجموعه راه حل نشان می‌دهد(Wang et al., 2017). شاخص کیفیت برای یکنواختی، توزیع یکنواختی یک مجموعه از راه‌حل‌ها را ارزیابی می‌کند و تغییرات فاصله بین راه‌حل‌ها را اندازه‌گیری می‌کند، همانند فاصله‌گذاری(Schott, 1995)[2]. شاخص قدرتمندی، یک راه حل نامغلوب متفاوت را به مجموعه‌ی راه‌حل‌ها اضافه می‌کند تا باعث پیشرفت ارزیابی شود. شاخص کیفیت برای انتشار و یکنواختی، به هم نزدیک می‌باشند و می‌توانند با یکدیگر استفاده شوند تا تنوع مجموعه راه‌حل‌ها را نشان دهند و به دو گروه دسته‌بندی می‌شوند:

شاخص‌های بر مبنای فاصله
شاخص‌های برپایِ تقسیم نواحی[3]

که یک ناحیه مخصوص را به تعداد زیادی سلول‌های هم‌اندازه پارتیشن‌بندی می‌کنند و سپس تعداد سلول‌هایی که دارای راه‌حل‌ها هستند را محاسبه می‌کنند. بعضی از آن‌ها، سلول‌ها را مانند توری‌هایی در نظر می‌گیرند که فضا را به تعدادِ زیادی فرا جعبه[4] پارتیشن‌بندی می‌کنند، همانند اندازه‌گیری تنوع[5](Goli et al., 2019). شاخص کیفیت برای تمام جنبه‌ها، همگرایی، انتشار، یکنواختی و اساسی را پوشش می‌دهد و به دو گروه دسته‌بندی می‌شوند:

شاخص کیفیت بر مبنای فاصله که فاصله پرتو نما نسبت به مجموعه راه‌حل‌های در نظر گرفته‌شده را اندازه‌گیری می‌نماید، همانند فاصله نسلی معکوس[6](Zhang et al., 2008; Zhou et al., 2005).

شاخص بر مبنای حجم[7] که اندازه‌ی حجم را اندازه‌گیری می‌نماید و به مجموعه راه‌حل‌های در نظر گرفته‌شده، اختصاص می‌یابد، همانند فرا حجم(Deb, 2001)[8].

جدول ۱ ارتباط بین معیارهای ارزیابی و گروه‌بندی آن‌ها را نشان می‌دهد.

جدول (۱) - ارتباط بین معیارهای ارزیابی و دسته‌بندی آن‌ها

دسته‌بندی	معیارهای ارزیابی					
	فرا حجم	فاصله نسلی معکوس	اندازه‌گیری تنوع	فاصله نسلی	تنوع خالص	فاصله‌گذاری
همگرایی	✓	✓		✓		
تنوع	✓	✓	✓		✓	
قدرتمندی	✓					
یکنواختی			✓			✓

در این تحقیق، شاخص فرا حجم، برای اندازه‌گیری راندمان نمونه‌های تست، به کار گرفته‌شده است و ۵۰۰ نقطه به‌صورت یکنواخت توزیع‌شده در هر نمونه، از پرتو نما، انتخاب می‌شود. برای ارزیابی عملکرد الگوریتم بهینه‌سازی چند هدفه تکاملی پیشنهادی،از تنظیمات پارامتر به شرح زیر استفاده‌شده است:

٤-۲- تنظیمات پارامترها

برای تمام الگوریتم‌های مورد مقایسه، تنظیمات پارامتری یکسانی از مسائل، به کار گرفته می‌شود. ۲۰ اجرای مستقل برای هر الگوریتم مقایسه‌ای بر روی هر یک از نمونه‌های تست، انجام می‌شود. شرط پایان برای هر الگوریتم، حداکثر ۱۰۰۰۰۰ ارزیابی برازش[9]، برای تمام نمونه‌های تست، در نظر گرفته می‌شود.
تست مجموع رتبه‌بندی ویلکسون[10]، برای مقایسه نتایج به‌دست‌آمده، در یک سطح اهمیت ۰،۰۵ از الگوریتم‌های آزمایش‌شده به کار می‌رود.

٤-۳- الگوریتم‌های مقایسه شده

اولین الگوریتم مقایسه‌ای، الگوریتم تکاملی چند هدفه دومین الگوریتم مقایسه‌ای، الگوریتم ژنتیک مرتب‌سازی نامغلوب است(Mkaouer et al., 2015) دومین الگوریتم مقایسه‌ای، الگوریتم تخمین چگالی چند هدفه بر پایه مدل منظم است(Zhang et al., 2008) و سومین الگوریتم مقایسه‌ای، الگوریتم تکاملی چند هدفه بر پایه مدل‌سازی معکوس با استفاده از فرآیند گوسین است(Cheng et al., 2015). قوی‌ترین الگوریتم‌های تکاملی به‌منظور مقایسه انتخاب‌شده‌اند. با توجه به این که الگوریتم پیشنهادی، زیر مجموعه روش‌های الگوریتم‌های تکاملی چند هدفه بر مبنای مدل می‌باشد، دو الگوریتم مقایسه‌ای از این گروه انتخاب‌شده‌اند (الگوریتم تخمین چگالی چند هدفه

1 Pure Diversity (PD
2 spacing
3 Region division based indicators
4 Hyper Box
5 Diversity Metric (DM)
6 Inversed Generational Distance (IGD)
7 Volume-based
8 Hyper Volume (HV)
9 Fitness Evalution (FE)
10 Wilcoxon rank sum test

بر پایه مدل منظم، الگوریتم تکاملی چند هدفه بر پایه مدل‌سازی معکوس با استفاده از فرآیند گوسین). همچنین الگوریتم پیشنهادی با یکی از بهترین الگوریتم‌های تکاملی چند هدفه سنتی (الگوریتم ژنتیک مرتب‌سازی نامغلوب-۲) مقایسه شده است.

٤-٤- توابع ارزیابی

روش پیشنهادی بر روی توابع تستِ ارائه‌شده توسط دب، تایل، توماس (دی تی آل زِد)[1] آزمایش‌شده است(Zhang et al., 2008). در این توابع تست، ارتباط بین متغیرهای تصمیم با استفاده از معادله (۲) می‌باشد:

$$\begin{cases} x_i \rightarrow \left(1+\alpha\frac{i}{n}\right)x_i \\ x_i^2 \rightarrow x_i^{\frac{1}{1+\beta\frac{i}{n}}} \end{cases} \quad (۲)$$

که در آن، i، اندیسِ هر متغیر تصمیم است و α و β، پارامترهای کنترلی هستند که $\alpha=3$ و $\beta=5$ است و تعداد متغیرهای تصمیم برابر با $n=30$ است.

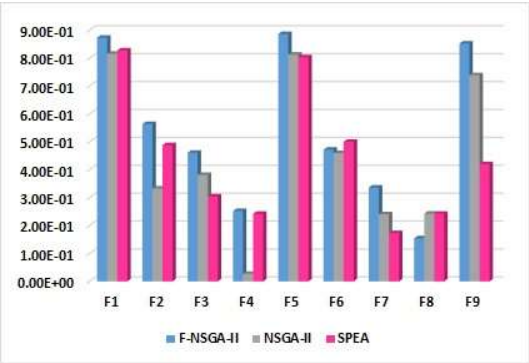
٥- یافته‌های تحقیق

در این بخش، در این بخش، نتایج آزمایش‌ها نشان داده می‌شود. نتایج با الگوریتم‌های مقایسه شده، بر روی توابع تست که در بخش قبل معرفی‌شده، آزمایش می‌شود و با استفاده از معیارهای شاخص‌های ارزیابی بیان‌شده در بخش قبل، بر روی سَکّوِ پِلَت ای‌ام اُ ارزیابی‌شده‌اند. از تست مجموع رتبه‌بندی ویلکسون، برای مقایسه نتایج به‌دست‌آمده، در یک سطح اهمیت ۰.۰۵ برای نمایش نتایج ارزیابی الگوریتم‌های آزمایش‌شده استفاده می‌شود. شکل (۳)، نتایج آماری بر روی الگوریتم‌های پیشنهادی و مقایسه شده، بر روی توابع تست را نشان می‌دهد.

نتایج شکل (۳) بر اساس شاخص ارزیابی فرا حجم می‌باشد. علت استفاده از این شاخص آن است که هر سه معیار همگرایی، تنوع و قدرتمندی را به‌طور هم‌زمان مورد ارزیابی قرار می‌دهد.

بر اساس این شاخص، معیارهای همگرایی، تنوع و قدرتمندی هر یک از الگوریتم‌های پیشنهادی و مقایسه‌ای مورد ارزیابی قرارگرفته و نتایج، در واقع برتری الگوریتم‌ها در هر یک از این معیارها به‌طور هم‌زمان را نشان می‌دهد.

با بررسی شکل (۳)، مشاهده می‌شود که الگوریتم پیشنهادی (F-NSGA-II) در ۶ تابع ارزیابی، عملکرد بهتری نسبت به الگوریتم تکاملی چند هدفه مبتنی بر مرتب سازی نامغلوب (NSGA-II) دارد. بنابراین طبق نتایج به‌دست‌آمده از شکل (۳)، مشاهده می‌شود که الگوریتم پیشنهادی عملکرد بهتری از لحاظ همگرایی، تنوع و قدرتمندی نسبت به سایر الگوریتم‌های مقایسه‌ای دارد. در واقع وقتی عدد حاصل از ارزیابی فراحجم در الگوریتم پیشنهادی بر روی توابع تست شده، نسبت به الگوریتم‌های مقایسه شده بیشتر است، به معنای کاهش تعداد ارزیابی و همگرایی بهتر در معیارهای همگرایی، تنوع و قدرتمندی می‌باشد. بنابراین روش پیشنهادی، الگوریتم تکاملی چند هدفه بر مبنای مدل را پیشنهاد می‌دهد که نمونه‌های ایجادشده از مدل، دارای تنوع، همگرایی و قدرتمندی بالاتری نسبت به روش‌های موجود می‌باشند و این امر به‌عنوان نقطه قوت الگوریتم پیشنهادی می‌باشد.



شکل (۳). نتایج شاخصِ مقادیر فرا حجم بر روی تابع‌های تست به‌منظور ارزیابی الگوریتم‌های مقایسه‌ای و پیشنهادی

٦- نتیجه گیری

در این مقاله، مطالعات تجربی نشان داده‌اند که در مجموع، روش ما بهتر از روش‌های SPEA و NSGA-II عمل می‌کند و با تکرار و ارزیابی عملکرد کمتر، نتایج بهتری به دست می‌آید .همچنین ما می توانیم به سرعت به راه حل های مورد نظر در مساله خود دسترسی پیدا کنیم که از ویژگی های اصلی این روش می باشد . عدم وابستگی این روش به پارامتر کنترل، به ویژه در خوشه بندی داده ها، از مزایای این روش نسبت به NSGA-II است.

مراجع

Babuka, R., Van der Veen, P., & Kaymak, U. (2002). Improved covariance estimation for Gustafson-Kessel clustering. 2002 IEEE World Congress on

1 Deb K, Thiele L, Laumanns M, Zitzler E (DTLZ)

Computational Intelligence. 2002 IEEE International Conference on Fuzzy Systems. FUZZ-IEEE'02. Proceedings (Cat .No. 02CH37291,(

Bosman, P. A., & Thierens, D. (2006). Multi-objective Optimization with the Naive $$\mathbb {M} $$ ID $$\mathbb {E} $$ A. In *Towards a New Evolutionary Computation* (pp. 123-157). Springer .

Cheng, R., He, C., Jin, Y., & Yao, X. (2018). Model-based evolutionary algorithms: a short survey. *Complex & Intelligent Systems*, *4*(4), 283-292 .

Cheng, R., Jin, Y., Narukawa, K., & Sendhoff, B. (2015). A multiobjective evolutionary algorithm using Gaussian process-based inverse modeling. *IEEE Transactions on Evolutionary Computation*, *19*(6), 838-856 .

Deb, K. (2001). *Multi-objective optimization using evolutionary algorithms* (Vol. 16). John Wiley & Sons .

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, *6*(2), 182-197 .

Fodor, I. K. (2002). *A survey of dimension reduction techniques* .

Goli, A., Zare, H. K., Tavakkoli-Moghaddam, R., & Sadegheih, A. (2019). Multiobjective fuzzy mathematical model for a financially constrained closed-loop supply chain with labor employment. *Computational Intelligence* .

Laumanns, M., & Ocenasek, J. (2002). Bayesian optimization algorithms for multi-objective optimization. International Conference on Parallel Problem Solving from Nature ,

Li, B., Li, J., Tang, K., & Yao, X. (2015). Many-objective evolutionary algorithms: A survey. *ACM Computing Surveys (CSUR)*, *48*(1), 1-35 .

Li, M., & Yao, X. Quality Evaluation of Solution Sets in Multiobjective Optimisation :A Survey .

Ma, H., Wei, H., Tian, Y., Cheng, R., & Zhang, X. (2021). A multi-stage evolutionary algorithm for multi-objective optimization with complex constraints. *Information Sciences*, *560*, 68-91 .

Mkaouer, W., Kessentini, M., Shaout, A., Koligheu, P ,. Bechikh, S., Deb, K., & Ouni, A. (2015). Many-objective software remodularization using NSGA-III. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, *24*(3), 1-45 .

Pantula, P. D., Miriyala, S. S., & Mitra, K. (2020). An evolutionary neuro-fuzzy C-means clustering technique. *Engineering Applications of Artificial Intelligence*, *89*, 103435 .

Pelikan, M., Sastry, K., & Goldberg, D. E. (2005). Multiobjective hBOA, clustering, and scalability. Proceedings of the 7th annual conference on Genetic and evolutionary computation ,

Schott, J. R. (1995). *Fault Tolerant Design Using Single and Multicriteria Genetic Algorithm Optimization* .

Škrjanc, I., Iglesias, J. A., Sanchis, A., Leite, D., Lughofer, E., & Gomide, F. (2019). Evolving fuzzy and neuro-fuzzy approaches in clustering, regression, identification, and classification: a survey. *Information Sciences*, *490*, 344-368 .

Smith, L. I. (2002). A tutorial on principal components analysis .

Van Veldhuizen, D. A., & Lamont, G. B. (1998). Evolutionary computation and convergence to a pareto front. Late breaking papers at the genetic programming 1998 conference ,

Wang, H., Jin, Y., & Yao, X. (2017). Diversity assessment in many-objective optimization. *IEEE Transactions on Cybernetics*, *47*(6), 1510-1522 .

Wang, Y., Xiang, J., & Cai, Z. (2012). A regularity model-based multiobjective estimation of distribution algorithm with reducing redundant cluster operator. *Applied Soft Computing*, *12*(11), 3526-3538 .

Zhang, Q., Zhou, A., & Jin, Y. (2008). RM-MEDA: A regularity model-based multiobjective estimation of distribution algorithm. *IEEE Transactions on Evolutionary Computation*, *12*(1), 41-63 .

پنجمین کنفرانس ملی انفورماتیک ایران
پژوهشگاه دانشهای بنیادی، پردیس فرمانیه، تهران
۱۳ و ۱۴ دی ماه ۱۴۰۲

زمان‌بندی آگاه به اوج توان در سامانه‌های بحرانی-مختلط سه سطحی چندهسته‌ای

شایان شکری[۱]، محراب طوقانی[۱]، ساره ملکی[۱]، سپیده صفری[۲]، شاهین حسابی[۳]
[۱] دانشجوی کارشناسی ارشد، دانشکده کامپیوتر، دانشگاه صنعتی شریف، تهران،
{shokri, sareh.maleki78, mehrab.toghani79}@sharif.edu
[۲] محقق پسادکتری، پژوهشکده علوم کامپیوتر، پژوهشگاه دانش‌های بنیادی، تهران
sepideh.safari@ipm.ir
[۳] دانشیار ، دانشکده مهندسی کامپیوتر کامپیوتر، دانشگاه صنعتی شریف، تهران
hessabi@sharif.edu

چکیده

امروزه با ظهور و گسترش سامانه‌های رایافیزیکی، توجه به طراحی سامانه‌های بحرانی-مختلط[۱] و حل چالش‌های این سامانه‌ها اهمیت ویژه‌ای یافته است. این سامانه‌ها از وظایف با چندین سطوح بحرانی مختلف تشکیل شده‌اند. با توجه به اجرای همزمان وظایف با درجه‌های بحرانی مختلف بر روی بستر سخت‌افزاری مشترک این سامانه‌ها، زمان‌بندی وظایف به صورتی که محدودیت‌های مختلف مانند زمان، دما و توان نقض نشود، امری حیاتی است. همچنین با پیشرفت فناوری ساخت، میزان توان مصرفی این سامانه‌ها نیز افزایش یافته است. افزایش توان مصرفی در سامانه‌های بحرانی-مختلط می‌تواند باعث افزایش دما و در نتیجه آسیب به سامانه شود. برای این منظور در این مقاله یک روش زمان‌بندی آگاه به اوج توان مصرفی برای سامانه‌های بحرانی-مختلط سه سطحی چند هسته‌ای ارائه شده است تا علاوه بر اجرای وظایف با درجه بحرانی زیاد[۲] پیش از موعد زمانی آن‌ها، توان سامانه نیز از حد مشخصی که توان طراحی حراتی[۳] نامیده می‌شود، فراتر نرود. در این مقاله، پس از ارئه‌ی روش نگاشت وظایف بر روی هسته‌ها و سپس تخصیص وظایف به هسته‌ها، به ارئه‌ی یک روش زمان‌بندی جدید برای وظایف بر اساس سطوح بحرانی آن‌ها پرداخته شده است.لازم به ذکر است هرگاه در حین زمان‌بندی، توان مصرفی سامانه بالاتر از توان حرارتی طراحی برود طبق مکانیزم‌های ارائه شده، اجرای وظیفه را متوقف کرده و از زمانی که این محدودیت نقض نشود اجرای آن از سر گرفته می‌شود. روش پیشنهادی ارائه شده با چند روش زمان‌بندی دیگر در سامانه‌های بحرانی-مختلط مقایسه شده است و نتایج به دست آمده نشان می‌دهد که روش پیشنهادی از نظر امکان زمان‌بندی و اجرای وظایف نسبت به روش‌های پیشین ٪۲۷ بهتر عمل می‌کند.

کلمات کلیدی

سامانه‌های بحرانی-مختلط سه سطحی، مدیریت اوج توان، توان حرارتی طراحی، نگاشت و زمان‌بندی وظایف بحرانی-مختلط

۱- مقدمه

در سامانه‌های بحرانی-مختلط، وظایف با درجه‌های بحرانی مختلف بر روی یک بستر سخت‌افزاری مشترک به منظور کاهش هزینه، مساحت و توان مصرفی اجرا می‌شوند [1]. این سامانه‌ها از چندین درجه بحرانی مختلف تشکیل شده‌اند که یکی از شناخته‌شده‌ترین نوع از این سامانه‌ها، سامانه‌های بحرانی-مختلط دو سطحی است، که در آن وظایف به دو دسته مختلف یعنی وظایف با درجه بحرانی زیاد و کم تقسیم می‌شوند. یکی از انواع پیچیده‌تر سامانه‌های بحرانی-مختلط ، سامانه‌های سه سطحی هستند که تا کنون در پژوهش‌های صورت گرفته توجه کمتری به آن شده است. در سامانه‌های بحرانی-مختلط سه‌سطحی وظایف به سه دسته مختلف وظایف با درجه بحرانی کم[۴]، متوسط[۵] و زیاد تقسیم می‌شوند که وظایف با درجه بحرانی کم دارای یک بدترین زمان اجرا به نام W^{LO}، وظایف با درجه بحرانی متوسط دارای دو بدترین زمان اجرای W^{LO} و W^{MI} و وظایف با درجه بحرانی زیاد دارای سه بدترین زمان اجرا می‌باشند که به ترتیب W^{LO}، W^{MI} و W^{HI} نامیده می‌شوند. سامانه در وضعیت عادی[۶] شروع به کار می‌کند که تمامی وظایف بر اساس W^{LO} اجرا می‌شوند. سپس در صورتی که وظیفه‌ای با درجه بحرانی متوسط یا وظیفه با درجه بحرانی زیاد به اندازه W^{LO} اجرا شود ولی به پایان نرسد، سامانه وارد وضعیت سرریز[۷] می‌شود. در این وضعیت هر دو نوع وظایف با درجه بحرانی متوسط و زیاد بر اساس W^{MI} اجرا می‌شوند. اگر یکی از وظایف با درجه بحرانی زیاد به اندازه W^{MI} اجرا شود ولی به پایان نرسد سامانه وارد وضعیت بحرانی می‌شود. در این وضعیت وظایف با درجه بحرانی زیاد به اندازه W^{HI} اجرا می‌شوند. در برخی از کارهای پیشین انجام شده در این حوزه، هنگامی که سامانه از حالت نرمال خارج می‌شود وظایف با درجه‌ی بحرانی پایین‌تر را از سامانه حذف کرده و اجرا نمی‌کنند [1]. اما در دیگر پژوهش‌ها یک مقدار مشخص به عنوان حداقل کیفیت خدمات برای وظایف با درجه‌ی بحرانی کم در نظر گرفته می‌شود [2].

[1] Mixed-Criticality Systems
[2] High-Critical Tasks
[3] Thermal Design Power (TDP)
[4] Low-Critical Tasks
[5] Middle-Critical Tasks
[6] Normal
[7] Overrun

با پیشرفت فناوری و افزایش تعداد هسته‌ها میزان توان مصرفی سامانه‌ها افزایش یافته است. افزایش توان مصرفی سامانه می‌تواند باعث افزایش دمای سامانه شود که این امر می‌تواند باعث آسیب جدی به سامانه شود. توان طراحی یک محدودیت توانی بر روی سامانه‌های بحرانی-مختلط می‌باشد که در صورتی که توان مصرفی سامانه از این حد مشخص فراتر رود دمای سامانه افزایش پیدا کرده و نهایاتا منجر به خرابی سامانه خواهد شد [3]. روش‌های مدیریت دمای پویا[1] برای مقابله با این مسئله است که از طریق افزایش سرعت خنک‌کننده‌ها و یا کاهش فرکانس کاری سامانه سعی در کاهش توان مصرفی و دمای آن را دارد. اما این روش‌ها موجب به کاهش چشمگیر عملکرد سامانه می‌شود [4]. همچنین کاهش فرکانس کاری سامانه می‌تواند باعث طولانی‌تر شدن زمان اجرای وظایف شود که با توجه به ماهیت ایمنی-بحرانی بودن سامانه‌های بحرانی-مختلط این مسئله می‌تواند باعث رعایت نکردن موعد زمانی مقرر و در نتیجه بروز فاجعه شود.

در این مقاله، یک روش زمان‌بندی آگاه به اوج توان برای سامانه‌های بحرانی-مختلط سه سطحی مبتنی بر قاب اجرای وظایف با در نظر گرفتن محدودیت‌های زمانی و توان طراحی تحت عنوان PPA-MiCs[2] ارائه شده است. روش پیشنهادی به این صورت است که ابتدا وظایف از بالاترین درجه بحرانی بر اساس بهره‌وری بر روی هسته‌ها نگاشت می‌شوند. سپس وظایف نگاشت شده بر روی هر هسته بر اساس متوسط توان مصرفی مرتب شده و بر اساس درجه بحرانی روی هسته‌ها اجرا می‌شوند. در صورتی که توان مصرفی سامانه از محدودیت توان حرارتی طراحی فراتر رود، اجرای وظیفه را متوقف کرده و از زمانی اجرای وظیفه را ادامه می‌یابد که این محدودیت نقض نشود. در نهایت از زمان‌های لختی[3] (بیکاری) ایجاد شده بر روی هر هسته برای بهبود کیفیت خدمات استفاده شده است.

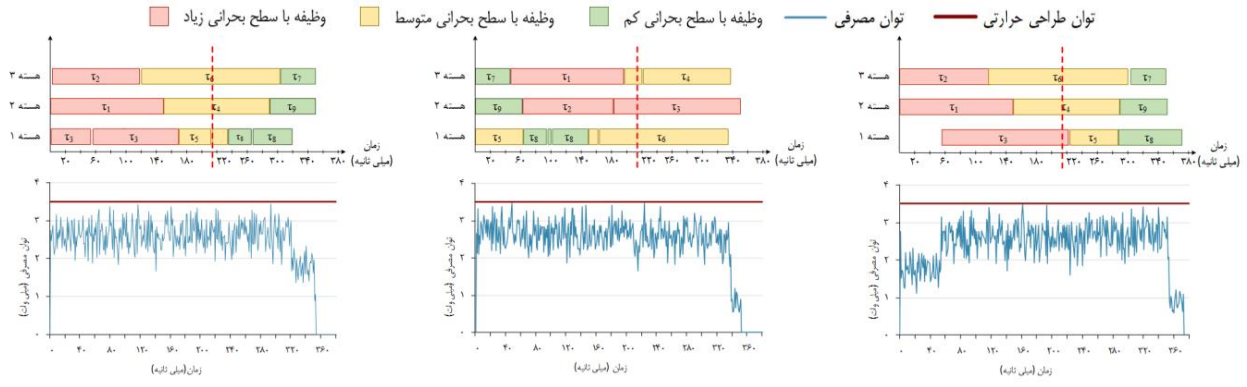
۱-۱- مثال انگیزشی

در شکل ۱ زمان‌بندی و توان مصرفی روش پیشنهادی PPA-MiCs و دو روش زمان‌بندی بدون اولویت سطوح بحرانی و روش انتقال وظایف از ابتدا در صورت نقض محدودیت توان حرارتی طراحی نشان داده شده‌اند. برای این منظور ۹ وظیفه با سطوح بحرانی مختلف بر روی سه هسته‌ی پردازشی نگاشت و زمان‌بندی شده‌اند. شکل ۱-الف و ۱-ب به ترتیب این دو روش را نشان می‌دهند. همانطور که مشهود است در صورتی که بخواهیم در این دو روش محدودیت توان حرارتی طراحی را رعایت کنیم امکان اجرای وظایف با درجه بحرانی زیاد پیش از موعد زمانی امکان‌پذیر نیست. در حالی که روش پیشنهادی PPA-MiCs که در شکل۱-ج نمایش داده شده است علاوه بر رعایت محدودیت توان حرارتی طراحی، محدودیت زمانی را نیز رعایت کرده است. در نتیجه مثال انگیزشی نشان می‌دهد که روش پیشنهادی PPA-MiCs در این مقاله نسبت به روش‌های دیگر برای حل مسئله‌ی زمان‌بندی وظایف بحرانی-مختلط سه‌سطحی بر روی سامانه‌های چندهسته‌ای با درنظرگرفتن محدودیت‌های زمانی و توان بحرانی حرارتی نتیجه بهتری را ارائه می‌دهد.

۲-۱- نوآوری‌ها

همان‌طور که مثال انگیزشی ذکر شده در بخش قبل نشان می‌دهد، درنظرگرفتن محدودیت‌های زمانی و توان حرارتی طراحی یک امر مهم در زمان‌بندی وظایف بحرانی-مختلط است. همچنین زمان‌بندی وظایف بحرانی-مختلط یک مسئله-ی حل‌نشدنی در زمان چندجمله‌ای بر حسب اندازه ورودی مسئله است. ما در این مقاله یک روش زمان‌بندی آگاه به اوج توان برای سامانه‌های بحرانی-مختلط سه سطحی چندهسته‌ای ارائه می‌دهیم که با درنظرگرفتن محدودیت‌های زمانی و توان حرارتی طراحی وظایف را بر روی هسته‌ها زمان‌بندی می‌کند. همچنین از زمان‌های لختی باقی‌مانده بر روی هسته‌ها برای بهبود کیفیت خدمات سامانه استفاده می‌کند. نوآوری‌های اصلی ارائه شده در این مقاله شامل موارد زیر است:

- درنظر گرفتن سامانه‌ی بحرانی-مختلط سه سطحی شامل وظایف با درجه‌ی بحرانی زیاد، متوسط و کم
- ارائه یک روش زمان‌بندی آگاه به اوج توان برای سامانه‌های بحرانی-مختلط سه‌سطحی
- ارائه‌ی روشی جهت بهبود کیفیت خدمات وظایف با درجه‌ی بحرانی کم و متوسط در سامانه با به کارگیری زمان‌های لختی موجود بر روی هسته‌ها



(ج) (ب) (الف)

شکل ۱: مثال انگیزشی: الف) روش انتقال وظایف از ابتد، ب) روش زمانبندی بدون اولویت سطوح بحرانی و ج) روش پیشنهادی PPA-MiCs

[1] Dynamic Thermal Management (DTM)
[2] Peak-Power-Aware Mixed-Criticality Systems (PPA-MiCs)
[3] Slack Time

جدول ۱: مروری بر کارهای پیشین

منبع	مدل سامانه			مدل معماری		مدیریت اوج توان	مدیریت توان متوسط
[5]	بی‌درنگ سخت		متناوب	چند هسته‌ای	همگن	✓	×
[6]					ناهمگن	✓	×
[4]			مبتنی بر قاب / متناوب	چند هسته‌ای	ناهمگن	✓	×
[14]			گراف جهت‌دار فاقد دور	تک هسته	-	×	✓
[7]	بی‌درنگ نرم		متناوب	چند هسته‌ای	همگن	✓	×
[8]			گراف جهت‌دار فاقد دور	چند هسته‌ای	همگن	✓	✓
[15]			مبتنی بر قاب	چند هسته‌ای	همگن	✓	✓
[9]	-		پراکنده	چند هسته‌ای	همگن	✓	×
[11]	بحرانی-مختلط	دو سطحی	متناوب	چند هسته‌ای	همگن	✓	×
[10]			گراف جهت‌دار فاقد دور	چند هسته‌ای	همگن	✓	×
[12]						✓	×
[13]		سه سطحی	متناوب	چند هسته‌ای	همگن	×	✓
روش پیشنهادی	بحرانی-مختلط	سه سطحی	مبتنی بر قاب	چند هسته‌ای	همگن	✓	×

در ادامه این مقاله در بخش ۲ کارهای پیشین این حوزه بررسی شده‌اند. سپس در بخش ۳ مدل سامانه و فرضیات در نظر گرفته شده در این مقاله بیان شده‌اند. در بخش ۴ روش پیشنهادی خود را به تفضیل ارایه داده و سپس در بخش ۵ روش پیشنهادی خود را با کارهای پیشین مقایسه می‌کنیم. در نهایت در بخش آخر از کار انجام شده در این مقاله نتیجه‌گیری به عمل می‌آوریم.

۲- کارهای پیشین

از آنجایی‌که این پژوهش بر مدیریت اوج توان مصرفی در زمان‌بندی سامانه‌های بحرانی-مختلط تمرکز دارد به همین دلیل، تحقیقات پیشین در این حوزه از دو حنبه‌ی زمان‌بندی‌های ارایه شده برای این سامانه‌ها و روش‌های مدیریت اوج توان مصرفی در سامانه‌های بی‌درنگ مورد بررسی قرار گرفته‌اند.

مدیریت اوج توان مصرفی باعث افزایش عمر باتری و افزایش قابلیت اطمینان سامانه‌ها می‌شود، از این رو پژوهش‌های [4][5][6][7][8][9] در این زمینه انجام شده‌اند که در برخی از این پژوهش‌ها صرفا به بحث زمان‌بندی وظایف و مدیریت اوج توان پرداخته شده است، مانند [5]. در تعدادی از این پژوهش‌ها علاوه بر پرداختن به مدیریت اوج توان مصرفی وزمان‌بندی وظایف، بر روی حفظ قابلیت اطمینان نیز تمرکز شده است [4][6][7] در پژوهش انجام شده در [4] روشی برای مدیریت توان حرارتی طراحی در سامانه‌های چندهسته‌ای غیر بحرانی-مختلط معرفی شده است. سپس تحقیقاتی بر روی همین محدودیت با درنظرگرفتن قابلیت اطمینان سامانه انجام شد. در پژوهش انجام شده در [5] دو روش زمان‌بندی تحت عنوان زودترین موعد زمانی نخست و زودترین موعد زمانی آخر به ترتیب برای زمان‌بندی وظایف روی هسته اصلی و هسته پشتیبان ارایه شده است. اخیراً در پژوهش انجام شده در [6] روشی برای مدیریت هم‌زمان دو محدودیت توان و قابلیت اطمینان در سامانه‌های چندهسته‌ای ناهمگن ارائه شده است. این روش از محدودیت توان حرارتی طراحی در سطح تراشه و محدودیت توان امن حرارتی در سطح هسته بهره می‌برد تا بتواند این دو متغیر مهم را هم‌زمان در مدیریت کند. در [7] نیز روش تنظیم پویای ولتاژ و فرکانس آگاه به قابلیت اطمینان و اوج توان مصرفی معرفی گردیده که از روش تکرار وظیفه برای تحمل‌پذیری اشکال در سامانه استفاده می‌کند. در پژوهش [8] روش زمان‌بندی برای جلوگیری از وقوع اوج توان و تداخل داده‌ها در میان وظایف ارائه شده است. از آنجایی که هیچ‌کدام از این پژوهش‌ها وظایف با سطوح بحرانی متفاوت را در نظر نگرفته‌اند، در ادامه برخی از پژوهش‌های این حوزه بررسی می‌شوند.

تمام پژوهش‌های انجام شده در حوزه سامانه‌های بحرانی-مختلط که مدیریت اوج توان را در نظر گرفته‌اند، به دلیل ساده‌تر بودن مدل‌سازی، سامانه‌هایی را با دو درجه بحرانی مورد بررسی قرار داده‌اند [10][11][12] برای حل این مسئله، پژوهش‌هایی انجام شده که وظایف را با بیش از دو درجه بحرانی در نظر گرفته‌اند اما هیچ‌کدام از آن‌ها مدیریت اوج توان در سامانه را درنظر نگرفته‌اند. برای مثال در [13] وظایف به صورت پویا به هسته‌ها تخصیص می‌یابند و تنظیم پویای ولتاژ و فرکانس نیز، برای کاهش مصرف انرژی به کار برده شده است. در این روش متوسط توان مصرفی به صورت یک محدودیت در نظر گرفته شده است اما توجهی به اوج توان در این کار نشده است. جدول ۱ بطور خلاصه کارهای پیشین انجام شده در این حوزه را نشان می‌دهد.

۳- مدل سامانه و فرضیات

در این بخش مدل سامانه و فرضیاتی که در این مقاله استفاده شده است معرفی می‌شوند.

۳-۱- مدل سامانه و وظایف

در این مقاله یک سامانه بحرانی-مختلط چندهسته‌ای متشکل از وظایف با سه درجه بحرانی مختلف در نظر گرفته شده است. وظایف بر روی این سامانه از سطوح مختلف استاندار DO-178B انتخاب می‌شوند [16]. وظایف بر روی این سامانه مبتنی بر قاب اجرای وظایف هستند بدین معنی که تمامی وظایف سامانه همزمان با هم آزادسازی شده و دارای یک موعد زمانی مشترک D هستند. هر وظیفه بر روی این سامانه با شش پارامتر مختلف ζ_i, , W_i^{LO}, W_i^{MI}, W_i^{HI}, L_i, O_i, معرفی می‌شود که به ترتیب درجه بحرانی وظیفه، بدترین زمان اجرای کوچک، بدترین زمان اجرای متوسط، بدترین زمان اجرای بزرگ، درجه بحرانی انتخابی از استاندار DO-178B و وضعیت عملیاتی وظیفه i ام است. همچنین هر وظیفه بر اساس اینکه به کدام یک از سطوح بحرانی سامانه تعلق دارد، می‌تواند تا سه مقدار بهره‌وری مختلف داشته باشد که به صورت زیر محاسبه می‌شود:

$$U^k = \sum_{\tau_i \in \Gamma}^{\zeta_i = k} u_i^k, \quad u_i^k = \frac{w_i^k}{D} \qquad (۱)$$

۲-۳- مدل توان

توان مصرفی سامانه بر اساس فرمول ۲ محاسبه می‌شود که شامل توان ایستا و پویا است. زمانی که وظیفه‌ای بر روی هسته‌ای در حال اجرا است توان مصرفی شامل توان پویا و ایستا است. اما در زمانی که وظیفه‌ای بر روی هسته در حال اجرا نباشد، توان مصرفی هسته تنها برابر با توان ایستا است.

$$P_{total} = P_{static} + P_{dynamic} = v.I_{leak} + \alpha.C_{eff}.v^2.f \qquad (۲)$$

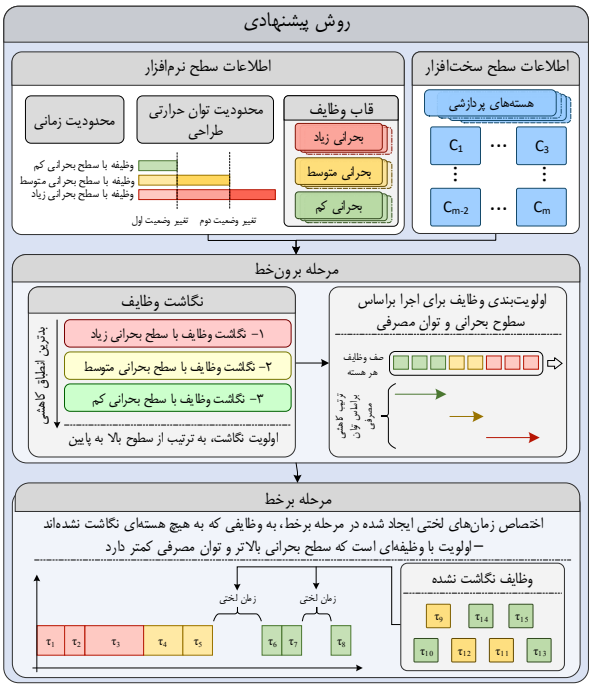
که در فرمول بالا α، C_{eff}، v و f به ترتیب بیانگر فاکتور فعالیت وظیفه، خازن مؤثر، ولتاژ و فرکانس هسته برای اجرای وظیفه و میزان جریان نشتی است.

۴- روش پیشنهادی

شکل ۲ روند کاری روش پیشنهادی PPA-MiCs را نمایش می‌دهد. ما یک سامانه بحرانی-مختلط سه سطحی با m هسته را در نظر گرفته‌ایم که n وظیفه مبتنی بر قاب اجرای وظایف بر روی آن اجرا می‌شود. هدف این روش پیشنهادی اجرای وظایف بر روی هسته‌ها به شکلی است که محدودیت‌های زمانی و توان حرارتی طراحی نقض نشود.

نگاشت وظایف بر روی هسته به‌صورت یک ماتریس M_{ij} نمایش داده می‌شود که i نشان‌دهنده وظیفه و j نشان‌دهنده هسته‌ی سامانه است. در صورتی که وظیفه i ام بر روی هسته j ام نگاشت شود مقداری برابر یک می‌گیرد و در غیر این صورت مقدار آن برابر صفر خواهد بود. همچنین توان مصرفی سامانه نیز به‌صورت یک ماتریس P_{it} نمایش داده می‌شود که بیانگر توان مصرفی وظیفه i ام در زمان t است. محدودیت زمانی در نظر گرفته شده برای سامانه را می‌توان به‌صورت زیر نمایش داد:

$$\forall j : \sum_i M_{ij}.wcet_i < Deadline \qquad (۳)$$



شکل ۲: نمای کلی روش پیشنهادی

الگوریتم ۱: روند نگاشت و زمانبندی وظایف

Inputs: $P = \{P_1, P_2, \dots, P_m\},\ \Gamma = \{\tau_1, \tau_2, \dots, \tau_n\},\ TDP$
Outputs: Mapping & Scheduling

1. $\Gamma' \leftarrow$ sort Γ based on criticality & utilization
2. **for** ($\tau_i \in \Gamma'$) **do**
3. Map τ_i on a prrcessor based on the worst-fit policy
4. **end for**
5. **if** (Eq. 5 is not satisfied) **do**
6. return NotFeasible
7. **end if**
8. **for** ($P_i \in P$) **do**
9. ${\Gamma'}_i \leftarrow$ Sort Γ based on criticality & average power
10. Schedule tasks in ${\Gamma'}_i$
11. **end for**
12. **while** (TDP is not satisfied at time t) **do**
13. $\tau_i \leftarrow$ select a task with lower start time which executed in time t
14. Suspend τ_i in time t & shift the remaning part of it
15. **end while**

همچنین محدودیت توان حرارتی طراحی نیز برای سامانه به‌صورت زیر بیان می‌شود:

$$\forall t : \sum_{ij} P_{it}.M_{ij} < TDP \qquad (۴)$$

۴-۱- نگاشت وظایف

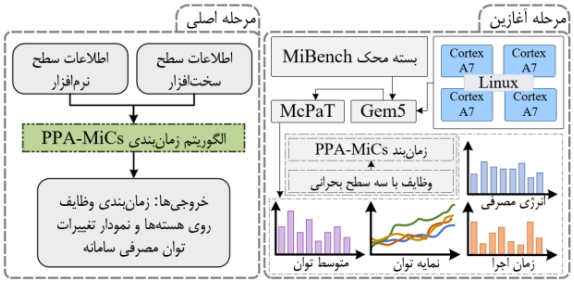
الگوریتم ۱، روش PPA-MiCs را نشان می‌دهد. برای نگاشت وظایف بر روی هسته‌ها ابتدا وظایف بر اساس سطوح بحرانی دسته‌بندی و سپس هر دسته بر اساس بهره‌وری به‌صورت نزولی مرتب می‌شود (خط ۱). پس از مرتب‌کردن وظایف، ابتدا وظایف با درجه بحرانی زیاد بر روی هسته‌ها نگاشت می‌شوند. برای نگاشت وظایف از سیاست بدترین انطباق کاهشی استفاده شده است. پس از نگاشت تمامی وظایف با درجه بحرانی زیاد بر روی هسته‌ها، به ترتیب وظایف با درجه بحرانی متوسط و وظایف با درجه بحرانی کم نیز به همین ترتیب بر روی هسته‌ها نگاشت می‌شوند (خط ۲-۴). باید توجه داشت که هنگام نگاشت وظایف با درجه بحرانی زیاد شرط زیر برقرار باشد (خط ۵):

$$\forall j : \sum_{i \in HI} M_{ij}.U(HI, HI)_i < 1 \qquad (۵)$$

در صورت برقرار بودن نامعادله بالا می‌توان نتیجه گرفت که سامانه در بدترین حالت می‌تواند تمامی وظایف با درجه‌ی بحرانی زیاد را بر روی هسته‌ها پیش از موعد زمانی اجرا کند. باید توجه داشت که شرط بالا تنها محدودیت زمانی را در نظر گرفته است در حالی که در این مقاله ما علاوه بر محدودیت زمانی، محدودیت توان حرارتی طراحی را نیز لحاظ کرده‌ایم.

۴-۲- زمان‌بندی وظایف

پس از نگاشت وظایف بر روی هسته‌ها، وظایف نگاشت شده بر روی هر هسته به ترتیب اولویت از لحاظ سطوح بحرانی و سپس وظایف هر دسته بر اساس توان مصرفی متوسط بصورت نزولی مرتب می‌شوند (خط ۹ در الگوریتم ۱). این کار به ما کمک می‌کند تا وظایف با بیشترین میزان توان مصرفی تا حد امکان در دورترین زمان نسبت به موعد زمانی خود اجرا شوند تا در صورتی که به دلیل نقض توان حرارتی طراحی مجبور به انتقال وظایف به سمت موعد زمانی آن‌ها شدیم، موعد زمانی از دست نرود. پس از مرتب نمودن وظایف در صف اجرای هسته‌ها وظایف باید اجرا شوند (خط ۱۰). در صورتی که در هر



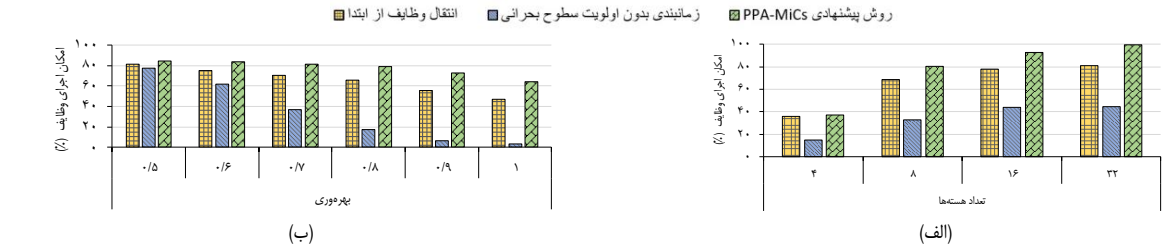
شکل ۵: ابزارهای استفاده شده در ارزیابی روش پیشنهادی

لحظه از زمان، توان مصرفی سامانه از توان حرارتی طراحی بالاتر رود اجرای وظیفه را متوقف کرده و از نقطه‌ای که اجرای مجدد آن باعث نقض توان حرارتی طراحی نشود، اجرای آن از سر گرفته می‌شود (خط ۱۳–۱۴). در صورتی که تمامی وظایف با درجه بحرانی زیاد به درستی پیش از موعد زمانی زمان‌بندی شوند و محدودیت توان حرارتی طراحی نقض نشود، اجرای وظایف بر روی سامانه امکان‌پذیر است.

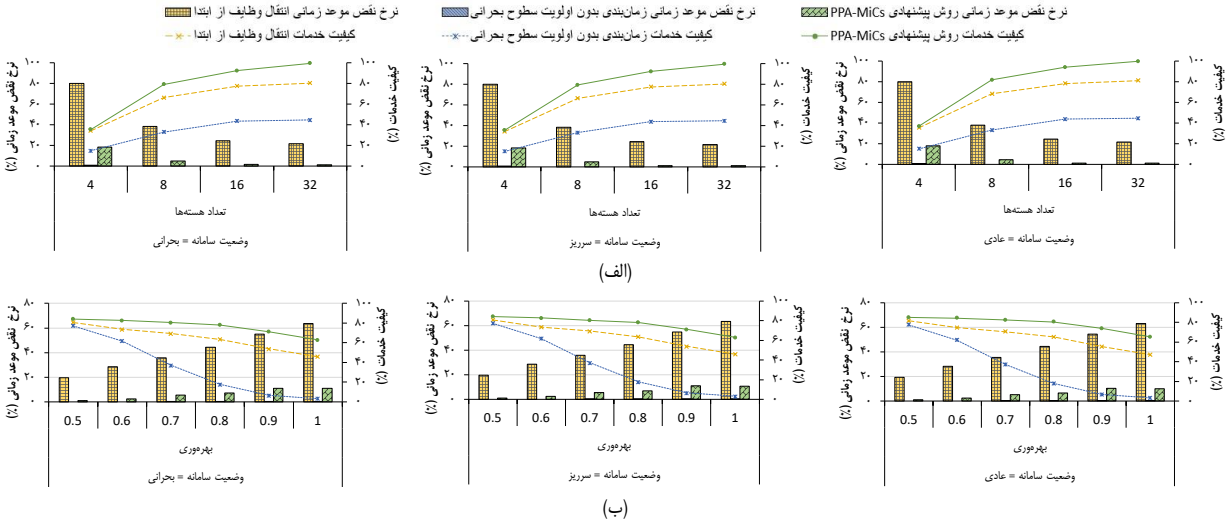
با توجه به شکل ۲، روش PPA-MiCs در مرحله برخط از زمان‌های لختی ایجاد شده برای اجرای وظایفی که در مرحله برون‌خط نگاشت نشده‌اند استفاده می‌کند. اگر اجرای وظیفه نگاشت نشده در زمان لختی ایجاد شده در پردازنده‌ای موجب نقض توان حرارتی طراحی سامانه نشود، آنگاه آن وظیفه در آن بازه زمانی و بر روی آن پردازنده زمان‌بندی و اجرا می‌شود. در روش‌های پویای پیشین شرط نقض توان حرارتی بررسی نشده است که این امر موجب رعایت نشدن اوج توان مصرفی سامانه می‌شود. سربار زمانی و محاسباتی روش PPA-MiCs در مرحله برخط به دلیل اینکه فقط از زمان‌های لختی برای اجرای وظایفی که نگاشت نشده‌اند استفاده می‌شود، بسیار ناچیز بوده و قابل صرف نظر کردن است و یا در بدترین حالت می‌توان آن را بخشی از بدترین زمان اجرای وظیفه در نظر گرفت [17].

۵- ارزیابی

در این بخش نتایج شبیه‌سازی روش پیشنهادی را بیان می‌کنیم. برای این منظور از دسته محک MiBench برای انتخاب وظایف واقعی برای شبیه‌سازی استفاده نموده‌ایم. اطلاعات وظایف شامل بدترین زمان اجرا، توان مصرفی پویا و ایستا مطابق پژوهش [10] بر روی پردازنده ARM Cortex-A7 شبیه‌سازی شده و بدست آمده است. در شکل ۳ طرح کلی از ابزارهای مورد استفاده در پیاده‌سازی روش پیشنهادی را نمایش می‌دهد. با توجه به اینکه این پژوهش اولین کار در این حوزه است که برای وظایف سه درجه بحرانی در نظر می‌گیرد، مقایسه آن با روش‌های پیشین، حتی آن‌هایی که دو درجه بحرانی در نظر گرفته‌اند، امکان‌پذیر نیست و در صورت مقایسه، نتایج عادلانه نخواهد بود. درنتیجه برای مقایسه، روش پیشنهادی خود را با روش انتقال وظایف از ابتدا (در صورت نقض توان حرارتی طراحی) و زمان‌بندی بدون در نظر گرفتن اولویت وظایف مقایسه نموده‌ایم. در روش انتقال وظایف از ابتدا در صورتی که محدودیت توان حرارتی طراحی نقض شود وظیفه از ابتدا به جلو منتقل می‌شود تا جایی که محدودیت توان حرارتی طراحی نقض نشود. روش PPA-MiCs از سه جنبه‌ی مختلف با این دو روش مقایسه شده است که عبارتند از امکان‌پذیری اجرا (که شامل زمان‌بندپذیری و رعایت محدودیت توان حرارتی طراحی است)، کیفیت خدمات و نرخ نقض موعد زمانی. امکان اجرای وظایف به معنی اجرای وظایف پیش از موعد زمانی و رعایت محدودیت توان حرارتی طراحی است. کیفیت خدمات و نرخ نقض موعد زمانی به ترتیب



شکل ۳: نتایج ارزیابی امکان اجرای وظایف روش پیشنهادی PPA-MiCs با روش‌های زمان‌بندی بدون اولویت سطوح بحرانی و انتقال وظایف از ابتدا



شکل ٤: نتایج ارزیابی کیفیت خدمات و نرخ نقض موعد زمانی روش پیشنهادی PPA-MiCs با روش‌های زمان‌بندی بدون اولویت سطوح بحرانی و انتقال وظایف از ابتدا

به صورت میزان اجرای وظایف با درجه بحرانی متوسط پیش از موعد زمانی و نرخ نقض موعد زمانی برای وظایف با درجه بحرانی کم تعریف می‌شود.

میزان بهره‌وری به ازای هر پردازنده را ۰.۵ تا ۱ (با گام ۰.۱) در نظر گرفته‌ایم. برای هر وظیفه درجه بحرانی آن را به صورت تصادفی و با توزیع یکنواخت از بین سه درجه بحرانی موجود انتخاب کرده‌ایم و وظایف را بر روی ۴، ۸، ۱۶ و ۳۲ هسته پردازشی همگن اجرا کرده‌ایم. همچنین سیستم را در سه وضعیت مورد بررسی قرار داده‌ایم: ۱) حالت عادی سامانه که همه هسته‌ها در وضعیت عادی هستند، ۲) سرریز که همه هسته‌ها در وضعیت سرریز هستند و ۳) بحرانی که در آن تمامی هسته‌ها در وضعیت بحرانی می‌باشند. همانطور که در شکل ۴-الف مشاهده می‌شود، امکان اجرای وظایف با افزایش بهره‌وری کاهش می‌یابد. با این حال، روش PPA-MiCs به دلیل انتقال بخشی از وظیفه به جای انتقال کل آن و استفاده بهتر از منابع پردازشی، نسبت به روش انتقال وظایف از ابتدا ۱۱.۵٪ بهبود دارد. همچنین روش PPA-MiCs به دلیل در نظر گرفتن سطوح بحرانی مختلف نسبت به روش زمان‌بندی بدون اولویت سطوح بحرانی ۴۳.۲۸٪ بهبود دارد. روند بهبود امکان اجرای وظایف با افزایش تعداد هسته‌ها به دلیل وجود منابع پردازشی بیشتر در شکل ۴-ب مشهود است.

در شکل ۵-الف و شکل ۵-ب، کیفیت خدمات و نرخ نقض موعد زمانی بررسی شده است. روش پیشنهادی به دلیل انتقال بخشی از وظیفه که باعث استفاده بهتر از منابع پردازشی می‌شود از نظر کیفیت خدمات نسبت به روش انتقال وظایف از ابتدا و روش زمان‌بندی بدون اولویت سطوح بحرانی به ترتیب، ۱۲.۳۳٪ و ۴۳.۲۳٪ بهبود یافته است. با افزایش تعداد هسته‌ها به دلیل وجود منابع پردازشی بیشتر نرخ نقض موعد زمانی و کیفیت خدمات نیز بهبود می‌یابد.

۶- نتیجه‌گیری و کارهای آتی

در این مقاله، یک روش زمان‌بندی آگاه به اوج توان برای سامانه بحرانی-مختلط سه‌سطحی مبتنی بر قاب اجرای وظایف بر روی بستر چندهسته‌ای با در نظر گرفتن محدودیت‌های زمانی و توان حرارتی طراحی ارائه شده است. در روش پیشنهادی وظایف بر اساس سطوح بحرانی بر روی هسته‌ها نگاشت می‌شوند. سپس بر اساس سطوح بحرانی و میانگین توان مصرفی، وظایف نگاشت شده بر روی هر هسته مرتب و سپس زمان‌بندی می‌شوند. در هر زمان که توان سامانه از توان حرارتی طراحی فراتر رود اجرای وظایف متوقف می‌شود و مجدداً از لحظه‌ای که محدودیت توان حرارتی طراحی قطعا رعایت خواهد شد اجرای باقی‌مانده‌ی زمان اجرای وظیفه انجام خواهد شد. نتایج مقایسه‌ی روش پیشنهادی PPA-MiCs نشان می‌دهد از جنبه‌ی زمان‌بندپذیری ۲۷٪ و از جنبه‌ی کیفیت خدمات ۲۸٪ نسبت به روش‌های دیگر بهبود داده شده است. در آینده روش ارائه شده در این مقاله بر روی سامانه‌های بحرانی-مختلط سه‌سطحی چندهسته‌ای ناهمگن توسعه می‌یابد. به این صورت که نگاشت وظایف بر روی هسته‌ها به صورتی انجام می‌پذیرد که توان مصرفی سامانه محدودیت توان حرارتی طراحی را نقض نکند.

مراجع

[1] A. Burns, and R. I. Davis, “A survey of research into mixed-criticality systems,” *Journal ACM Computing Surveys*, volume 50, p-p1-37, 2018.

[2] H. Su and D. Zhu, "An Elastic Mixed-Criticality task model and its scheduling algorithm," *2013 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, Grenoble, France, 2013, pp. 147-152 .

[3] Intel Corporation. 2007. Dual-core intel xeon processor 5100 series datasheet, revision 003.

[4] W. Munawar, H. Khdr, S. Pagani, M. Shafique, J. -J. Chen and J. Henkel, "Peak Power Management for scheduling real-time tasks on heterogeneous many-core systems," *2014 20th IEEE International Conference on Parallel and Distributed Systems (ICPADS)*, Hsinchu, Taiwan, 2014, pp. 200-209.

[5] M. Ansari, A. Yeganeh-Khaksar, S. Safari and A. Ejlali, "Peak-Power-Aware Energy Management for Periodic Real-Time Applications," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems,* vol. 39, no. 4, pp. 779-788, April 2020.

[6] M. Ansari, M. Pasandideh, J. Saber-Latibari and A. Ejlali, "Meeting Thermal Safe Power in Fault-Tolerant Heterogeneous Embedded Systems," in *IEEE Embedded Systems Letters*, vol. 12, no. 1, pp. 29-32, March 2020.

[7] A. Yeganeh-Khaksar, M. Ansari and A. Ejlali, "ReMap: Reliability Management of Peak-Power-Aware Real-Time Embedded Systems Through Task Replication," in *IEEE Transactions on Emerging Topics in Computing*, vol. 10, no. 1, pp. 312-323, 1 Jan.-March 2022.

[8] B. Lee, J. Kim, Y. Jeung and J. Chong, "Peak power reduction methodology for multi-core systems," *2010 International SoC Design Conference*, Incheon, Korea (South), 2010, pp. 233-235.

[9] Z. Lee, B. Yun and K. G. Shin, "Reducing Peak Power Consumption inMulti-Core Systems without ViolatingReal-Time Constraints," in *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 4, pp. 1024-1033, 2014.

[10] S. Safari, H. Khdr, P. Gohari-Nazari, M. Ansari, S. Hessabi and J. Henkel, "TherMa-MiCs: Thermal-Aware Scheduling for Fault-Tolerant Mixed-Criticality Systems," in *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 7, pp. 1678-1694, 1 July 2022.

[11] M. Navardi, B. Ranjbar, N. Rohbani, A. Ejlali and A. Kumar, "Peak-Power Aware Life-Time Reliability Improvement in Fault-Tolerant Mixed-Criticality Systems," in *IEEE Open Journal of Circuits and Systems*, vol. 3, pp. 199-215, 2022.

[12] B. Ranjbar, A. Hosseinghorban, M. Salehi, A. Ejlali and A. Kumar, "Toward the Design of Fault-Tolerance-Aware and Peak-Power-Aware Multicore Mixed-Criticality Systems," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 41, no. 5, pp. 1509-1522, May 2022.

[13] H. Sobhani, S. Safari, J. Saber-Latibari, and S. Hessabi, “REALISM: Reliability-aware energy management in multi-level mixed-criticality systems with service level degradation,” *Journal of Systems Architecture*, vol. 117, p. 102090, Aug. 2021.

[14] Z. Li, S. Ren, and G. Quan, “Energy minimization for reliability-guaranteed real-time applications using DVFS and checkpointing techniques,” *Journal of Systems Architecture*, vol. 61, no. 2, pp. 71–81, Feb. 2015.

[15] M. Ansari, M. Salehi, S. Safari, A. Ejlali and M. Shafique, "Peak-Power-Aware Primary-Backup Technique for Efficient Fault-Tolerance in Multicore Embedded Systems," in *IEEE Access*, vol. 8, pp. 142843-142857, 2020.

[16] DO-178C. 2011. Software considerations in airborne systems and equipment certification. RTCA, Inc.

[17] Buttazzo, Giorgio C, “Hard real-time computing systems: predictable scheduling algorithms and applications,” *Springer Science & Business Media,* vol. 24, 2011.

پنجمین کنفرانس ملی انفورماتیک ایران
پژوهشگاه دانشهای بنیادی، پردیس فرمانیه، تهران
۱۳ و ۱۴ دی ماه ۱۴۰۲

روش پیش‌واکشی داده کارا در پردازنده‌های گرافیکی

صبا مستوفی[۱]، هاجر فلاحتی[۲]، نگین ماهانی[۳]، پژمان لطفی کامران[۴]، حمید سربازی آزاد[۵]

[۱] دانشجو ، دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف، تهران،
s.mostofi.virgo@gmail.com

[۲] استادیار پژوهشکده کامپیوتر-پژوهشگاه دانش‌های بنیادی، تهران،
hfalahati@ipm.ir

[۳] استادیار گروه مهندسی کامپیوتر مجتمع آموزش عالی زرند، دانشگاه شهید باهنر، کرمان،
negin.mahani@uk.ac.ir

[۴] دانشیار پژوهشکده کامپیوتر-پژوهشگاه دانش‌های بنیادی ، تهران،
plotfi@ipm.ir

[۵] استاد دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف و پژوهشکده کامپیوتر-پژوهشگاه دانش‌های بنیادی، تهران
azad@sharif.edu

چکیده

پردازنده‌های گرافیکی با به‌کارگیری سلسله مراتب حافظه و موازات سطح نخ، سعی در پنهان‌سازی تاخیر دسترسی به حافظه‌ی برون تراشه دارند. اما در عمل به دلیل محدود شدن موازات سطح ریسماا، میزان توانایی آن‌ها در پنهان‌سازی تاخیر دسترسی به حافظه‌ی برون تراشه کاهش می‌یابد. پیش‌واکشی یکی از راه‌های موثر در کاهش تاخیر دسترسی به حافظه می‌باشد. در زمینه پیش‌واکشی در پردازنده‌های گرافیکی، پژوهش‌های محدودی در گذشته صورت گرفته است و نتایج نشان از تاثیرگذاری مثبت پیش‌واکشی در بهبود کارایی پردازنده‌های گرافیکی دارد. با این وجود، این روش‌ها به خوبی فضای موجود برای پیش‌واکشی داده‌ها را پوشش نداده‌اند.

در این پژوهش، روشی نوین در زمینه پیش‌واکشی داده در پردازنده‌های گرافیکی ارائه می‌گردد. پیش‌واکشی در این روش، امکان شناسایی فواصل گامی بین دستورات دسترسی به حافظه مختلف را فراهم می‌آورد و می‌تواند زنجیره‌ای از آدرس‌هایی که در آینده مورد نیاز ریسمان‌ها می‌باشد را پیش‌واکشی کند. روش ارائه شده می‌تواند ۸۰ درصد درخواست‌های دسترسی به حافظه را با دقت بالای ۹۰ درصد پیش‌واکشی کند. در نهایت به‌کارگیری این روش باعث بهبود ۱۷ درصدی کارایی پردازنده‌های گرافیکی شده و مصرف انرژی را نیز تا ۱۷ درصد کاهش می‌دهد.

کلمات کلیدی

پردازنده گرافیکی، حافظه روی تراشه، پیش‌واکشی، کارایی.

۱- مقدمه

در سال‌های اخیر، محدودیت در افزایش فرکانس پردازنده‌ها، تبدیل به چالشی اصلی برای پیروی از قانون مور شده بود. از این‌رو استفاده از تراشه‌های چندهسته‌ای تبدیل به راهکاری برای افزایش کارایی گردید و طراحان سامانه‌های کامپیوتری به رویکرد پردازش موازی تعداد زیادی نخ روی آوردند. با پیشرفت فناوری، امروزه پردازنده‌های گرافیکی همه منظوره به شکل گسترده‌ای در اجرای برنامه‌های محدود به حافظه مورد استفاده قرار می‌گیرند. این برنامه ها، زمینه‌های مختلفی از یادگیری ماشین گرفته تا محاسبات علمی را پوشش می‌دهند و امروزه کاربردهای بیشماری در زمینه‌های مختلف مانند پزشکی، نظامی، مالی، هواشناسی، بیولوژی، اقتصادی وابزارهای طراحی خودکار دارند. ویژگی اصلی این برنامه‌ها، محدود بودن کارایی آن‌ها به تعداد تعاملاتشان با حافظه می‌باشد. روند رو به رشد اندازه‌ی مجموعه‌های کاری این قبیل برنامه‌ها، باعث افزایش تعداد درخواست‌های دسترسی به حافظه شده و در نتیجه، افزایش سرعت دسترسی به حافظه در پردانده‌های گرافیکی اهمیت ویژه‌ای پیدا کرده است [2] ,[1] .

به منظور سرویس‌دهی بهتر به درخواست‌های حافظه و همچنین بهره برداری از مجاورت داده در برنامه‌های عام منظوره‌ی محدود به حافظه، پردازنده‌های گرافیکی از یک ساختار سلسله مراتبی حافظه همانند ساختار استفاده شده برای واحد پردازش مرکزی استفاده کرده‌اند. این ساختار از حافظه‌ی نهان سطح یک، حافظه‌ی نهان سطح دو و همچنین حافظه‌ی اصلی برون تراشه تشکیل شده است. حافظه‌ی نهان سطح یک، به صورت محلی و به ازای هر چندپردازنده‌ی جریانی قرار داده شده و هدف آن پشتیبانی از دسترسی‌های محلی به حافظه با سرعت بالا است. یک حافظه‌ی نهان سطح دو برای تمامی چندپردازنده‌های جریانی به صورت مشترک در دسترس است و هدف آن کاهش تأخیر دسترسی به حافظه‌ی اصلی می‌باشد. اگر درخواست‌ها در حافظه‌ی نهان سطح دو نیز یافت نشوند، باید تأخیر بسیار زیادی را برای دسترسی به حافظه‌ی اصلی تحمل کنند [3] .

در کنار استفاده از ساختار سلسله مراتبی حافظه، ساختار اصلی پردازنده‌های گرافیکی بر مبنای اجرای همزمان تعداد زیادی نخ به صورت موازی است که به آن موازات سطح نخ می‌گویند. انتظار می‌رود پردازنده‌ی گرافیکی با توجه به دارا بودن موازات سطح نخ بالا و هم‌چنین امکان تعویض نسبتا سریع، بتواند تأخیر دسترسی به حافظه را تا حد زیادی پنهان کند [4]. اما پژوهش‌ها نشان داده است که بالا بودن موازات سطح نخ، گاهاً می‌تواند برعکس عمل کرده و کارایی پردازنده‌های گرافیکی را کاهش دهد. تعداد زیاد درخواست‌های دسترسی به حافظه که توسط این نخ‌ها ارسال می‌شود تأخیر دسترسی به

حافظه را افزایش می‌دهد و باعث کاهش کارایی حافظه‌های نهان می‌شود [5], [6].

با توجه به محدودیت‌های موجود در پردازنده‌های گرافیکی، اندازه‌ی حافظه نهان سطح یک به نسبت تعداد بالای نخ‌های در حال اجرا، بسیار کوچک است. در نتیجه، این حافظه‌ی نهان سطح یک، فضای کافی برای قرار دادن تمامی بلوک‌های مورد نیاز نخ‌ها را ندارد. به همین دلیل، درخواست‌های جدیدی که توسط سایر نخ‌ها به حافظه‌ی نهان سطح یک ارسال می‌شوند، می توانند باعث اخراج بلوک‌های دیگر موجود در حافظه‌ی نهان سطح یک گردند. تاثیر منفی این رخداد می‌تواند تا حدی زیاد شود که تمامی نخ‌های در حال اجرا، به دلیل نیاز به دسترسی به حافظه متوقف شوند و پردازنده‌ی گرافیکی سیکل‌های ساعت زیادی را به دلیل تأخیر دسترسی به حافظه بیکار شود. چنین حالتی دیوار حافظه نام دارد که می‌تواند مشکلات جدی از جمله عدم استفاده مناسب از منابع، افزایش توان مصرفی و کاهش کارایی پردازنده‌ی گرافیکی را به همراه داشته باشد [7].

تاکنون پژوهش‌های پیشین، روش‌های مختلفی را برای حل مشکل دیوار حافظه ارائه کرده‌اند [8]–[10].یکی از این راه‌کارها که نتایج بسیار مثبتی را در کاهش تأخیر دسترسی به حافظه به همراه داشته است، روش پیش‌واکشی می‌باشد. در این روش تلاش می‌شود با بررسی الگوهای دسترسی به حافظه، دسترسی های آینده پیش‌بینی شده و زودتر به سطوح بالاتر حافظه آورده شوند. پیش‌واکشی، اولین بار برای واحد پردازش مرکزی مطرح و کارایی آن مورد بررسی قرار گرفت [11], [12].پژوهش های اخیر دریافتند که با توجه به ویژگی‌های ساختاری پردازنده‌های گرافیکی و همچنین ویژگی‌های برنامه‌های آن‌ها، پیش‌واکشی می‌تواند به عنوان روشی مؤثر در پردازنده‌های گرافیکی نیز به کار برده شود و نتایج مثبتی بر کارایی و توان مصرفی داشته باشد [13], [14].

در این پژوهش، تمرکز ما روی بهبود کارایی سلسله مراتب حافظه‌ی استفاده شده در پردازنده‌های گرافیکی با استفاده از روش پیش‌واکشی می‌باشد. بررسی‌های ما نشان داده است که کارایی تمامی روش‌های پیشین ارائه شده در این حوزه، به دلیل تمرکز بر یادگیری یک فاصله‌ی گامی ثابت میان درخواست‌های دسترسی به حافظه، محدود شده است و امکان بهبود درصد پوشش این روش‌ها وجود دارد. ما با بررسی الگوی دسترسی به حافظه در روش‌های مختلف، دریافتیم که در کنار یک فاصله‌ی گامی ثابت، می‌توان یک زنجیره از این فواصل را هم ذخیره کرد. سپس با توجه به ساختار یک دستور چندین داده در پردازنده‌های گرافیکی، این الگوی زنجیره‌ای در سطوح مختلف به تعداد زیاد تکرار می‌شود و به ما در پیش‌بینی الگوهای دسترسی آینده کمک خواهد کرد. روش پیش‌واکشی پیشنهاد شده در این پژوهش، *Snake* نام دارد و به دنبال یافتن یک زنجیره از فواصل گامی دسترسی به حافظه در سطوح مختلف می‌باشد.

با توجه به نتایج شبیه‌سازی با استفاده از شبیه‌ساز *Accel-sim* روی بارهای کاری عام منظوره‌ی مختلف در بردازنده‌ی گرافیکی، *Snake* می تواند ۸۰ درصد درخواست‌های آینده را پیش‌بینی کند و ۷۵ درصد آن‌ها را به موقع به مصرف حافظه‌ی نهان برساند. هم‌چنین با استفاده از سازوکار پیش‌واکشی پیشنهاد شده، کارایی پردازنده‌گرافیکی به طور میانگین ۱۷ درصد بهبود یافته و ۱۷ درصد انرژی مصرفی کل کاهش می‌یابد. نوآوری‌های این مقاله به شرح زیر است:

• ما ناکارآمدی مکانیسم‌های پیش‌واکشی اخیرا ارائه شده در فراهم کردن داده‌های پیش‌واکشی دقیق برای برنامه‌های *GPGPU* محدود به حافظه را نشان می‌دهیم.

• ما *Snake* را معرفی می‌کنیم؛ یک مکانیزم پیش‌واکشی میان–نخی کارآمد بر مبنای زنجیره‌های گام که در سطوح مختلف (درون–نخی و میان–نخی) عمل می‌کند.

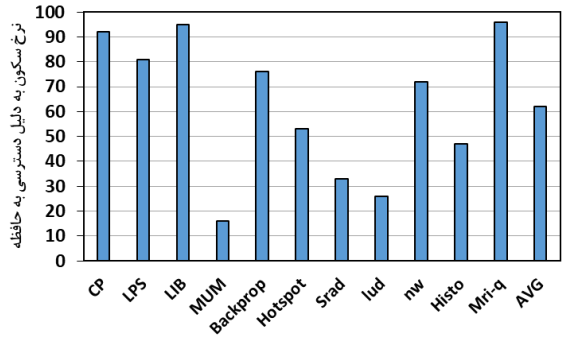
• ما نشان می‌دهیم که *Snake* باعث بهبود عملکرد برنامه‌های *GPGPU* محدود به حافظه به میانگین ۱۷٪ شده و مصرف انرژی را تا ۱۷٪ کاهش می‌دهد.

• ما نشان می‌دهیم که *Snake* پوشش و دقت به مراتب بهتری دارد (تا ۶۰٪ و ۵۵٪ به ترتیب) نسبت به برترین مکانیسم پیش‌واکشی معاصر (*CTA*-Aware[14])

۲- انگیزه و روش‌های پیشین

با توجه به رشد محبوبیت و کاربردهای گسترده‌ی پردازنده‌های گرافیکی در انجام محاسبات برنامه‌های کاربردی مختلف، بهبود کارایی و کاهش مصرف انرژی در آن‌ها از اهمیت ویژه‌ای برخوردار است. با توجه به بالا بودن تعداد نخ‌های موازی در حال اجرا در پردازنده‌های گرافیکی و کوچک بودن نسبی فضای حافظه‌ی نهان سطح یک، بهره‌وری حافظه‌ی نهان سطح یک کاهش می‌یابد. در نتیجه، تأثیر مثبت ساختار سلسله مراتبی در کاهش تأخیر دسترسی به حافظه محدود می‌شود.

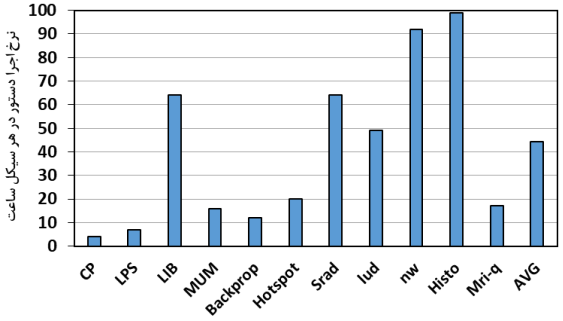
شکل (۱) تعداد سیکل‌های ساعتی که در هنگام اجرای یک برنامه، تمام کلاف‌های پردازنده‌ی گرافیکی به‌دلیل درخواست دسترسی به حافظه در حالت سکون قرار می‌گیرند را نسبت به کل تعداد سیکل‌های ساعتی که پردازنده‌ی گرافیکی در هنگام اجرای همان برنامه در حالت سکون قرار گرفته است نشان می‌دهد. بر طبق شکل (۱)، به‌طور میانگین، ۶۲ درصد مواقعی که پردازنده‌ی گرافیکی در حالت سکون قرار دارد، به‌دلیل تأخیر دسترسی به حافظه می‌باشد.



شکل (۱) : نرخ تعداد سیکل‌های ساعت که تمام کلاف‌های پردازنده‌ی گرافیکی به دلیل تأخیر دسترسی به حافظه در حالت سکون قرار دارند در مقایسه با تعداد کل سیکل‌های ساعتی که پردازنده‌ی گرافیکی به دلایل مختلف در حالت سکون قرار دارد.

به‌علاوه، شکل (۲) افزایش تعداد دستور اجرا شده به‌ازای هر سیکل ساعت را در حالتی که تأخیر دسترسی به حافظه صفر باشد، نشان می‌دهد. در حقیقت در این حالت فرض شده است که داده‌ی مورد نیاز، از حافظه بدون صرف زمان

در اختیار حافظه‌ی نهان سطح یک قرار می‌گیرد و در نتیجه در اختیار نخ‌ها قرار می‌گیرد تا عملیات خود را انجام دهند. در این حالت مشاهده می‌شود که در صورت حذف تاخیر دسترسی به حافظه، کارایی پردازنده‌ی گرافیکی به طور میانگین تا ۴۰٪ افزایش می‌یابد.



شکل (۲) : نرخ بهبود تعداد دستور اجرا شده به ازای هر سیکل ساعت در حالتی که تأخیر دسترسی به حافظه حذف شود.

باتوجه به این نتایج، با وجود بالا بودن موازات سطح نخ در پردازنده‌های گرافیکی، هم‌چنان تأخیر دسترسی به حافظه به‌عنوان یک گلوگاه در پردازنده‌های گرافیکی مطرح می‌شود و کاهش آن می‌تواند به بهبود کارایی پردازنده‌های گرافیکی منجر شود.

پیش‌واکشی به‌عنوان یک روش مؤثر در کاهش تأخیر دسترسی به حافظه مطرح می‌شود. پیش‌واکشی پیش از این در حوزه‌ی واحد پردازش مرکزی مورد بررسی قرار گرفته است. روش‌های پیش‌واکشی سخت‌افزاری ارائه شده برای پردازنده‌های گرافیکی را می‌توان به دو دسته پیش‌واکش‌هایی آگاه از اصل محلیت داده و پیش‌واکش‌های گامی تقسیم بندی کرد. تمامی روش‌هایی که تاکنون برای پیش‌واکشی آگاه از محلیت داده ارائه شده‌اند، مشکل پیش‌واکشی بیش از حد و غیر دقیق را دارا می‌باشند. اما روش‌های پیش‌واکشی گامی، از آن‌جایی که داده‌ای که هر نخ باید روی آن پردازش انجام دهد، از روی مقادیر ثابت متناظر با همان نخ شامل ابعاد بلوک آن نخ، شناسه‌ی بلوک آن نخ و شناسه‌ی نخ تعیین می‌شود. این ویژگی سبب می‌شود تا دسترسی به حافظه توسط نخ‌ها دارای الگوی گامی ثابت و قابل پیش‌بینی باشد. از طرفی، از آن‌جایی که به دلیل معماری یک دستور چندین نخ، این الگو در سطوح مختلف تکرار می‌شود. و با یادگیری آن در یک سطح می‌توان از اطلاعات یادگرفته شده برای پیش‌واکشی در سطوح دیگر استفاده کرد.

پژوهش‌های پیشین سازوکارهای پیش‌اکشی مبتنی بر گام مخصوص ویژگی‌های پردازنده‌های گرافیک را از قبیل پیش‌واکشی درون–نخی، میان–نخی و میان–بلوک نخ را معرفی کرده‌اند. پیش‌واکشی با استفاده از فاصله‌ی گامی درون–نخی [13] به دقت و پوشش بالا در حلقه‌های عمیق دست می‌یابد، اما با در ساختار پردازنده‌های گرافیکی حلقه‌های عمیق با پردازش موازی جایگزین شده‌اند که باعث افت درصد پوشش روش درون_نخی می‌شود. پیش‌واکشی میان–نخی [13] به دلیل نزدیک بودن زمان‌بندی اجرای نخ‌های مجاور به دلیل نبود زمان کافی باری پیش‌واکشی از مشکل عدم دقت رنج می‌برد. در روش پیش‌واکشی میان–بلوک نخ [14]، از آنجایی که فاصله‌ی زمان‌بندی اجرای بلوک‌های نخ زیاد است مشکل کمبود زمان برای پیش‌واکشی را حل می‌کند اما از طرفی به دلیل زمان‌بر بودن محاسبه‌ی آدرس پایه‌ی هر بلوک نخ، درصد پوشش بسیار کمی دارد.

۳- مطالب اصلی

هدف ما در این بخش، معرفی سازوکار شناسایی رشته‌ها در روش پیش‌واکشی *Snake* است که علاوه بر فواصل گامی سطوح مختلف، زنجیره‌ای از فواصل میان دستورات بارگیری از حافظه را نیز محاسبه کرده و با توجه به آن‌ها پیش‌واکشی را انجام می‌دهد. در این پژوهش، برای ذخیره‌سازی اطلاعات از دو جدول با نام‌های *Head* و *Tail* استفاده می‌شود. با استفاده از این جدول‌ها، امکان شناسایی، آموزش و استفاده از این زنجیره‌ها در سطوح مختلف مهیا می‌شود.

جدول*Head* نخستین جدولی است که برای ذخیره‌سازی اطلاعات از آن استفاده می‌شود. این جدول به صورت مستقیم با واحد بارگیری/ذخیره‌سازی در ارتباط است و اطلاعات مورد نیاز را از این واحد دریافت می‌کند. در این جدول سه مورد مهم شناسه‌ی کلاف، آدرس درخواست شده، و شمارنده‌ی برنامه نگه‌داری می‌شوند. هدف استفاده از این جدول، ذخیره‌ی آخرین شمارنده‌ی برنامه و آدرس درخواست شده‌ی متناظر با آن در هر کلاف است.

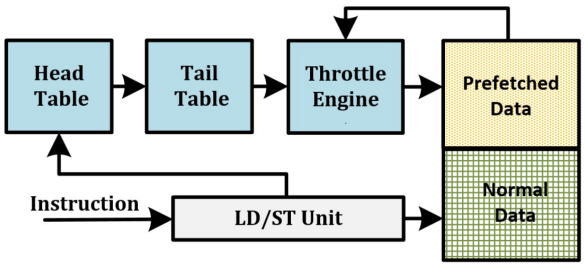
جدول *Tail* با استفاده از شمارنده‌های برنامه اندیس‌گذاری شده است و اطلاعات مربوط به فاصله‌های گامی را نگه می‌دارد. بررسی‌های ما نشان می‌دهد که علاوه بر فاصله‌ی میان دستورات بارگیری مختلف، فواصل درون–کلافی و میان–کلافی نیز حائز اهمیت هستند و نیاز داریم هرکدام را به صورت مستقل ذخیره کنیم چرا که امکان پیدا کردن رابطه‌ی مشخص میان آن‌ها وجود ندارد. درنتیجه، هر درایه‌ی جدول باید شامل اطلاعات فواصل گامی در سطوح مختلف باشد. در دو ستون اول این جدول اطلاعات مربوط به دو شمارنده‌ی برنامه‌ی پشت سر هم ذخیره می‌شود. در ستون سوم فاصله‌ی گامی میان این دو شمارنده‌ی برنامه ذخیره می‌شود. ستون چهارم یک بردار از داده‌های باینری به طول تعداد کلاف‌ها به نام آرایه‌ی شناسه‌ی کلاف می‌باشد. ستون‌های پنجم و ششم به ترتیب فاصله‌های گامی میان–کلافی و درون–کلافی را ذخیره می‌کنند. ستون‌های هفتم و هشتم نیز برای ذخیره‌سازی اطلاعات مربوط به وضعیت آموزش فواصل گامی درون–نخی و درون–کلافی استفاده می‌شوند.

در حالت کلی، سازوکار اجرای برنامه به صورت یک دستور چندین نخ که در پردازنده‌های گرافیکی وجود دارد، باعث می‌شود کلاف‌های مختلف یک دستور یکسان را روی داده‌های مختلف توسط نخ‌های خود اجرا کنند. با این حال، با جلو رفتن در برنامه، عوامل مختلفی می‌توانند بر ترتیب اجرای دستورات در هر کلاف اثرگذار باشند. عواملی مانند رخداد سکون در خط لوله‌ی اجرای دستورات به دلایلی همچون وجود دستور پرش، می‌تواند ترتیب اجرای دستورات در کلاف‌های مختلف را بر هم بزند. درنتیجه، برای ذخیره کردن هر رشته، باید شناسه‌ی کلافی که این رشته را مشاهده کرده است نیز لحاظ شود. برای مشخص کردن شناسه‌ی کلاف‌هایی که دستورات متناظر با دو شمارنده‌ی برنامه در ستون‌های اول و دوم را اجرا کرده و فاصله‌ی گامی ستون سوم را مشاهده کرده‌اند، از آرایه‌ی شناسه‌ی کلاف استفاده می‌شود.

با توجه به مشاهدات ذکر شده مشخص است که پیش‌واکشی در حافظه‌ی نهان سطح یک، می‌تواند اثرات مخربی را به همراه داشته باشد. درنتیجه باید سازوکاری ارائه شود تا داده‌های پیش‌واکشی شده اثری روی داده‌های موجود داخل حافظه‌ی نهان سطح یک نداشته باشند. در معماری‌های جدیدتر ارائه شده برای پردازنده‌های گرافیکی (مانند معماری *Volta*)، فضای حافظه‌ی نهان سطح یک و حافظه‌ی اشتراکی یکی شده‌اند. این تغییرات کار ما را برای

قرار دادن داده‌های پیش‌واکشی شده در حافظه‌ی اشتراکی آسان‌تر می‌کند چراکه می‌توان بخش مشترک حافظه‌ی بین حافظه‌ی نهان سطح یک و حافظه‌ی اشتراکی را به دو قسمت تقسیم کرد و داده‌های پیش‌واکشی شده را در فضای جداگانه‌ای از این بخش حافظه‌ی اشتراکی ذخیره کرد. *Snake* داده‌های پیش‌واکشی شده را در قسمت بالای حافظه‌ی نهان یکپارچه و داده‌های اصلی حافظه‌ی نهان سطح یک را در قسمت پایین ذخیره می‌کند و همواره اولویت را به ذخیره‌سازی داده‌های حافظه‌ی نهان سطح یک می‌دهد به گونه‌ای که اطمینان حاصل کند داده‌های پیش‌واکشی شده هرگز باعث اخراج داده‌های مفید موجود در حافظه‌ی نهان سطح یک نمی‌شوند. بدین منظور دقت پیش‌واکشی مورد بررسی قرار می‌گیرد. تا زمانی که دقت پیش‌واکشی کم‌تر از ۸۰٪ باشد، اولویت اخراج از حافظه‌ی یکپارچه همیشه با داده‌های پیش‌واکشی شده است.

در صورتی که پیش‌واکشی باعث ایجاد محدودیت در فضای ذخیره‌سازی حافظه‌ی نهان یکپارچه و یا پهنای باند موجود بین سطوح مختلف حافظه در پردازنده‌های گرافیکی گردد، سازوکار متوقف‌سازی پیش‌واکش فعال می‌شود. در صورت عدم وجود فضای خالی در حافظه‌ی یکپارچه، پیش‌واکش برای ۵۰ سیکل ساعت متوقف می‌شود تا هم به داده‌های اصلی موجود در حافظه‌ی نهان و هم به داده‌های پیش‌واکشی شده فرصت کافی داده شود تا استفاده شوند. پس از آن با توجه به دقت پیش‌واکشی بین داده‌های اصلی و داده‌های پیش‌واکشی شده کاندیدهای اخراج انتخاب می‌شوند و فضای آزاد شده مجدد مورد استفاده‌ی هر دو بخش قرار می‌گیرد. در صورت مواجه با نزدیک شدن به ۷۵٪ مصرف پهنای باند تئوری، پیش‌واکشی خاموش می‌شود تا از فشار بر روی بخش‌های مختلف حافظه کاسته شود و مجدد پیش‌واکش فعالیت خود را از سر بگیرد. شکل (۳) سازوکار سطح بالای *Snake* را نشان می‌دهد.



شکل (۳) : معماری سطح بالای *Snake*

۴- ارزیابی

در این بخش ابتدا محیط شبیه‌سازی و برنامه‌های محک استفاده شده در این پژوهش را معرفی می‌کنیم. در قدم بعدی، به ارزیابی روش پیش‌واکشی پیشنهادی،Snake ، در مقایسه با حالت پایه و روش‌های موجود می‌پردازیم.

۴-۱- محیط شبیه‌سازی

در این مطالعه، عملکرد و مصرف انرژی Snake با استفاده از Accel-Sim v1.2.0 ارزیابی می‌شود. به‌طور خاص، مصرف انرژی Snake با استفاده از Accel-Wattch v1.0 تجزیه و تحلیل می‌شود.

۴-۲- برنامه‌های محک

برای ارزیابی و مقایسه روش پیش‌واکشی ارائه شده از سه دسته بارکاری استفاده شده است. بارکاری مرسوم به ispass از ۱۲ برنامه‌ی کاربردی تشکیل شده است که همگی سطح بالایی از تعداد نخ‌های موازی دارند و به پهنای باند حساس می‌باشند. همچنین، رخداد نزاع در این برنامه‌ها زیاد می‌باشد. دسته‌ی دوم برنامه‌های کاری مرسوم به Rodinia می‌باشد. کارایی برنامه‌های این بارکاری، محدود به حافظه است. دسته‌ی سوم بارهای کاری، مرسوم به Parboil می‌باشد. برنامه‌های این بار کاری، توان عملیاتی پردازنده‌های گرافیکی را بررسی می‌کنند.

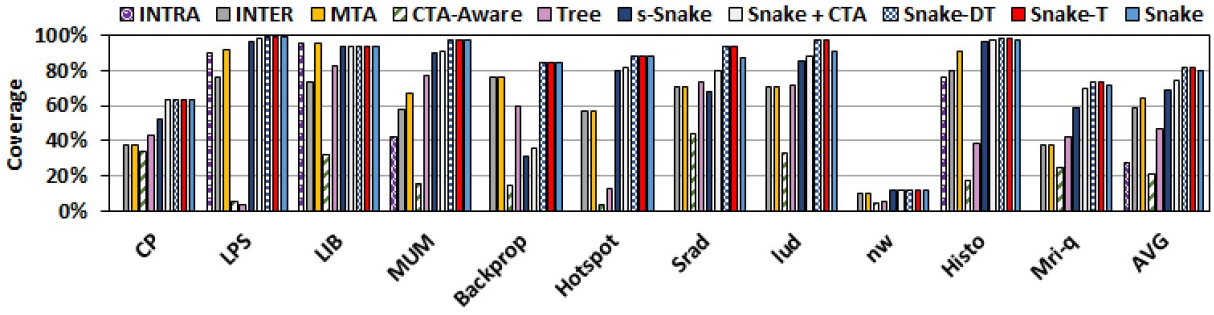
۴-۳- پارامترهای طراحی

در بررسی عملکرد پیش‌واکشی، دو عامل نقش اساسی دارند: پوشش ارائه شده توسط پیش‌واکشی و دقت داده‌های پیش واکشی شده. پوشش به معنای میزان آدرس‌های درخواست شده در کل برنامه است که توسط پیش‌واکشی به درستی تشخیص داده شده است. در طرف دیگر، دقت به معنای بخشی از آدرس‌های درست تشخیص داده شده توسط پیش‌واکشی است که در حافظه‌ی نهان استفاده شده‌اند.

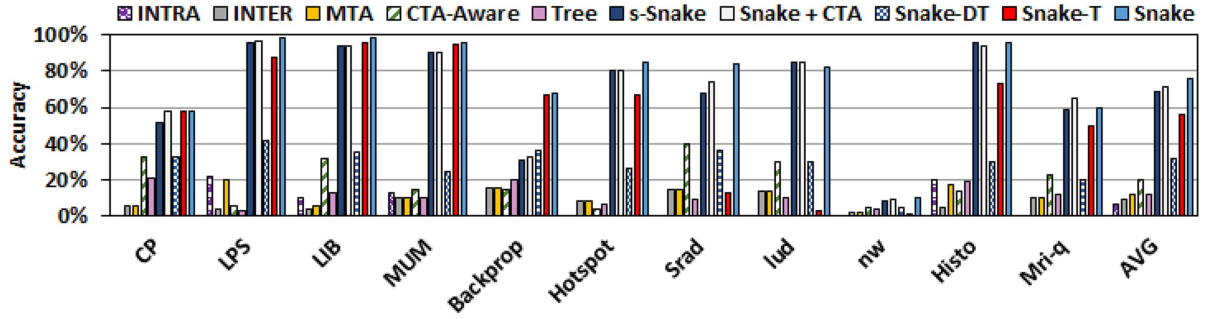
۴-۴- پارامترهای ارزیابی و سربار

برای ارزیابی سربار بخش‌های محاسباتی روش پیش‌واکشی*Snake* و سایر روش‌های پیشین *Synopsys Design Compiler* استفاده کردیم. هم‌چنین برای محاسبات سربار بخش‌های حافظه از ابزار *Cacti*کرده و با استفاده از اطلاعات استخراج شده‌ی مربوط به بخش‌های مختلف در شبیه‌ساز *Accel-sim*، پارامترهای طراحی را محاسبه کردیم. با توجه به خروجی *Cacti* روش پیش‌واکشی *Snake* به ازای هر دسترسی به حافظه‌ی نهان، ۶.۴ پیکو ژول انرژی مصرف می‌کند.

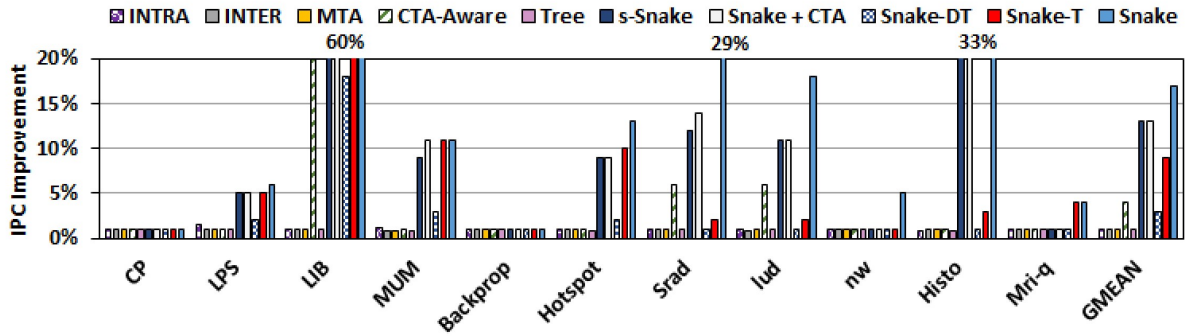
برای محاسبه‌ی سربار حافظه‌ی مورد نیاز برای ذخیره‌سازی جداول روش پیش‌واکشی با تعداد ۱۰ درایه برای جدول *Tail*، روش پیش‌واکشی *Snake* در مقایسه با روش CTA-Aware، ۲۰ بایت سربار فضای ذخیره‌سازی دارد اما درصد پوشش و دقت بسیار بالاتری را ارائه می‌دهد.



شکل (٤) : مقایسه‌ی درصد پوشش Snake و سایر روش‌ها



شکل (٥) : مقایسه‌ی درصد دقت Snake و سایر روش‌ها



شکل (٦) : مقایسه‌ی درصد بهبود کارایی Snake و سایر روش‌ها

۵-۴- نتایج شبیه‌سازی

میزان پوشش و دقت روش‌های پیش‌واکشی یکی از معیارهای اساسی در مقایسه‌ی این روش‌ها می‌باشد. شکل (۴) درصد پوشش روش پیش‌واکشی *Snake* را در مقایسه با سایر روش‌ها نشان می‌دهد.

همان‌طور که در شکل (۴) مشخص است، در بسیاری از برنامه‌ها درصد پوشش روش پیش‌واکشی درون-کلافی صفر است و امکان تشخیص هیچ الگویی را ندارد. روش پیش‌واکشی آگاه از چندین نخ، بالاترین درصد پوشش را در میان روش‌های پیشین ارائه شده برای پیش‌واکشی با استفاده از فاصله‌ی گامی در پردازنده‌های گرافیکی دارد. همان‌طور که شکل (۴) نشان می‌دهد، استفاده از فاصله‌ی گامی درون-نخی به تنهایی نیز درصد پوشش بالایی دارد. با این‌حال، ترکیب آن با پیش‌واکشی درون-کلافی و میان-کلافی، درصد پوشش بالاتری را ارائه می‌دهد. روش پیش‌واکشی آگاه از بلوک نخ، به طور کل درصد پوشش کمی دارد. به دلیل تطبیق‌پذیر بودن روش پیش‌واکشی

Snake با سایر روش‌های پیشین، امکان استفاده موازی از *Snake* در کنار سایر روش‌ها وجود دارد. به کارگیری این روش در کنار پیش‌واکشی با استفاده از فاصله گامی درون–نخی ، می‌تواند سبب بهبود درصد پوشش این روش شود. به طور میانگین، روش پیش‌واکشی *Snake*، امکان پوشش ۸۰ درصد درخواست‌های دسترسی به حافظه را دارد.

شکل (۵) درصد دقت پیش‌واکشی در روش‌های مختلف را نشان می‌دهد. زمان ارسال درخواست پیش‌واکشی و محدودیت فضای ذخیره‌سازی حافظه‌ی نهان سطح یک، دو عامل اصلی در کاهش دقت پیش‌واکشی هستند. همان‌طور که شکل (۵) نشان می‌دهد، روش پیش‌واکشی *Snake* دقت قابل قبولی در مقایسه با روش‌های پیشین دارد و به طور میانگین ۷۵ درصد درخواست‌های آینده را به موقع به حافظه‌ی نهان سطح یک می‌رساند. بالا بودن دقت پیش‌واکشی *Snake* در مقایسه با روش‌های قبلی دو دلیل اساسی دارد:

اولاً، همان‌طور که پیش‌تر گفته شد، در پیش‌واکشی میان–نخی، زمان کافی برای واکشی داده از حافظه وجود دارد. این نکته در بررسی میزان دقت روش پیش‌واکشی میان–نخی *s-Snake* نیز مشخص است. با توجه به شکل (۵)، در روش پیش‌واکشی میان–نخی، به طور متوسط، بیش از ۹۵ درصد داده‌هایی که درست پیش‌واکشی شده‌اند، توسط حافظه‌ی نهان مصرف می‌شوند. ثانیاً، در کنار دقت بالای پیش‌واکشی، جداسازی فضای ذخیره‌سازی داده‌های پیش‌واکشی شده از فضای داده‌های موجود در حافظه‌ی نهان سطح یک، این امکان را فراهم می‌کنند تا داده‌های پیش‌واکشی شده، مدت زمان بیشتری در حافظه باقی بمانند و مشکل فضای محدود ذخیره‌سازی نیز بدین ترتیب برطرف می‌شود.

همان‌طور که گفته شد، پیش‌واکشی دقیق می‌تواند درنهایت به بهبود کارایی پردازنده‌های گرافیکی منجر شود. یکی از معیارهای اندازه‌گیری بهبود کارایی، بررسی نرخ اجرای دستور در واحد سیکل ساعت است. شکل (۶) نرخ اجرای دستور در واحد سیکل ساعت که به حالت پایه بدون پیش‌واکشی نرمال‌سازی شده را نشان می‌دهد. با توجه به شکل (۶) روش پیش‌واکشی *Snake* می‌تواند نرخ اجرای دستورات در واحد سیکل ساعت را نسبت به حالت پایه‌ی بدون پیش‌واکشی تا حداکثر ۶۰ درصد روی بار کاری LIB و به طور متوسط تا ۱۷ درصد بهبود دهد و از این طریق کارایی پردازنده‌های گرافیکی را افزایش دهد. نکته‌ی قابل توجه در مورد روش‌های پیش‌واکشی *INTRA*{و *INTER*، اثر منفی آن‌ها بر روی کارایی سیستم از طریق کاهش نرخ اجرای دستورات در واحد سیکل ساعت نسبت به حالت بدون پیش‌واکشی می‌باشد. دلیل این اثر منفی، پایین بودن دقت در این روش‌ها می‌باشد. روش *MTA* نیز، به عنوان یک روش ترکیبی، مشکلات هر دو روش قبلی را دارد.

۵- مراجع

[1] N. Nematollahi, M. Sadrosadati, H. Falahati, M. Barkhordar, and H. Sarbazi-Azad, "Neda: Supporting direct inter-core neighbor data exchange in GPUs," *IEEE Computer Architecture Letters*, vol. 17, no. 2, pp. 225–229, 2018.

[2] S. Kamil, A. Cheung, S. Itzhaky, and A. Solar-Lezama, "Verified lifting of stencil computations," *SIGPLAN Not.*, vol. 51, no. 6, pp. 711–726, Aug. 2016, doi: 10.1145/2980983.2908117.

[3] N. Nematollahi *et al.*, "Efficient Nearest-Neighbor Data Sharing in GPUs," *ACM Trans. Archit. Code Optim.*, vol. 18, no. 1, pp. 1–26, Mar. 2021, doi: 10.1145/3429981.

[4] C. NVIDIA, "C Programming Guide: Design Guide," PG-02829-001 v6. 5, NVIDIA, Santa Clara, Calif, USA, 2014.

[5] P. Xiang, Y. Yang, and H. Zhou, "Warp-level divergence in GPUs: Characterization, impact, and mitigation," in *2014 IEEE 20th International Symposium on High Performance Computer Architecture (HPCA)*, IEEE, 2014, pp. 284–295. Accessed: Dec. 01, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/6835939/

[6] H. Wang, F. Luo, M. Ibrahim, O. Kayiran, and A. Jog, "Efficient and fair multi-programming in GPUs via effective bandwidth management," in *2018 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, IEEE, 2018, pp. 247–258. Accessed: Dec. 01, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8327013/

[7] S. Rajbhandari, O. Ruwase, J. Rasley, S. Smith, and Y. He, "ZeRO-infinity: breaking the GPU memory wall for extreme scale deep learning," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, St. Louis Missouri: ACM, Nov. 2021, pp. 1–14. doi: 10.1145/3458817.3476205.

[8] I. N. Wehn, "Wednesday Keynote: The Memory Wall: Challenges and Solutions," in *2019 32nd IEEE International System-on-Chip Conference (SOCC)*, IEEE, 2019, pp. 1–2. Accessed: Dec. 01, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9088072/

[9] J. Hong, S. Cho, and G. Kim, "Overcoming Memory Capacity Wall of GPUs With Heterogeneous Memory Stack," *IEEE Computer Architecture Letters*, vol. 21, no. 2, pp. 61–64, 2022.

[10] S. Darabi *et al.*, "OSM: Off-chip shared memory for GPUs," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 12, pp. 3415–3429, 2022.

[11] M. Shakerinava, F. Golshan, A. Ansari, P. Lotfi-Kamran, and H. Sarbazi-Azad, "State-of-the-art data prefetchers," in *Advances in Computers*, vol. 125, Elsevier, 2022, pp. 55–67. Accessed: Dec. 01, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0065245821000814

[12] G. Ayers, H. Litz, C. Kozyrakis, and P. Ranganathan, "Classifying Memory Access Patterns for Prefetching," in *Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems*, Lausanne Switzerland: ACM, Mar. 2020, pp. 513–526. doi: 10.1145/3373376.3378498.

[13] J. Lee, N. B. Lakshminarayana, H. Kim, and R. Vuduc, "Many-thread aware prefetching mechanisms for GPGPU applications," in *2010 43rd Annual IEEE/ACM International Symposium on Microarchitecture*, IEEE, 2010, pp. 213–224. Accessed: Dec. 01, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/5695538/

[14] G. Koo, H. Jeon, Z. Liu, N. S. Kim, and M. Annavaram, "Cta-aware prefetching and scheduling for gpu," in *2018 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, IEEE, 2018, pp. 137–148. Accessed: Dec. 01, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8425168/

پنجمین کنفرانس ملی انفورماتیک ایران
پژوهشگاه دانشهای بنیادی، پردیس فرمانیه، تهران
۱۳ و ۱۴ دی ماه ۱۴۰۲

شتاب‌دهنده مبتنی بر پردازش درون حافظه برای شبکه‌های عصبی ژرف

هاجر فلاحتی[۱]، نگین ماهانی[۲]

[۱] استادیار پژوهشکده کامپیوتر-پژوهشگاه دانش‌های بنیادی ، تهران،
hfalahati@ipm.ir

[۲] استادیار گروه مهندسی کامپیوتر مجتمع آموزش عالی زرند، دانشگاه شهید باهنر، کرمان،
negin.mahani@uk.ac.ir

چکیده

یکی از چالش‌های اصلی در پردازش الگوریتم‌های یادگیری ماشین تنگنای پهنای باند حافظه است. شتاب‌دهنده های درون حافظه پتانسیل رفع این مشکل را دارند. با این حال، راه حل مبتنی بر شتاب‌دهنده ی درون حافظه برای رسیدگی به این مشکل با دو چالش رو به رو است. اول این که، شتاب-دهنده‌ی درون حافظه باید از مجموعه ی بزرگی از الگوریتم‌های یادگیری ماشین پشتیبانی کند. دوم این که، راه‌حل باید به اندازه کافی کارآمد باشد تا از پهنای باند مناسب بهره‌مند شود در حالی‌که محدودیت‌های توان و مساحت لایه‌ی منطقی حافظه ی پشته‌ای سه‌بعدی را رعایت کند.

*در این مقاله ما یک شتاب‌دهنده‌ی ناهمگن (به نام **PUZZLE**) پیشنهاد می‌کنیم که ابتدا الگوهای پردازشی مشابه در الگوریتم‌های یادگیری ماشین را شناسایی می‌کند. در ادامه برای هر الگو یک واحد پردازشی پیشنهاد می‌کند. واحدهای پردازشی در لایه‌ی منطقی یک حافظه‌ی پشته‌ای سه بعدی قرار می‌گیرند و از پهنادی بالای این حافظه‌ها بهره می‌برند. علاوه بر این واحدهای پردازشی، مکانیزم پیشنهادی امکان اتصال به واحدهای پردازشی همه‌منظوره مانند پردازنده مرکزی و پردازنده گرافیکی را دارد. نتایج ارزیابی در ۱۲ الگوریتم معروف یادگیری ماشین نشان می دهد که مکانیزم پیشنهادی از شتاب دهنده‌های پیشرفته امروزی با حافظه پشته‌ای سه بعدی از نظر کارایی و بهره‌وری انرژی به ترتیب ۱.۵ برابر و ۲۹ برابر بهتر عمل می کند.*

کلمات کلیدی

یادگیری ماشین، شبکه عصبی ژرف، فاز آموزش، انعطاف‌پذیری، دیوار حافظه، پردازش درون حافظه.

۱- مقدمه

یادگیری ماشین امروزه در کاربردهای مختلفی، مانند بازی تا پزشکی استفاده می‌شود. این استفاده گسترده از الگوریتم‌های یادگیری ماشین منوط به ارائه بسترهایی با عملکرد بالا برای آموزش مدل‌ها در طول فاز آموزش هستند. سپس در فاز استنتاج مدل آموزش دیده، برای ارزیابی داده‌هایی که تا به حال ندیده است استفاده می‌شود. آموزش مدل‌های یادگیری ماشین به طور قابل‌توجهی حجم پردازشی بالایی دارد و درعین حال، فشار زیادی نیز بر حافظه وارد می‌کند (نیاز به پهنای باند بالا برای انتقال حجم بالای داده‌ها دارند که به عنوان دیوار حافظه شناخته می‌شود) [۱-۱۸]. از سوی دیگر مدل‌های یادگیری ماشین، مانند شبکه‌های عصبی عمیق، درحال پیچیده‌تر شدن هستند. با توجه به این ویژگی‌ها، شتاب دهنده‌ی درون حافظه [۶-۲۰] یک راه حل مناسب برای حل چالش دیوار حافظه و تسریع الگوریتم‌های یادگیری ماشین است.

با ظهور حافظه‌های سه بعدی پشته‌ای [۲۱-۲۵]، پردازش درون حافظه‌ای [۶،۸،۹،۱۱،۱۲،۱۴-۲۰] به یک راه حل قابل پیاده‌سازی تبدیل شد. کارهای بسیاری، شتاب‌دهنده‌های درون حافظه را برای الگوریتم‌های یادگیری ماشین طراحی کرده‌اند. به علت محدودیت بودجه مساحت و توان یک حافظه‌ی پشته‌ای سه بعدیی، بیشتر شتاب‌دهنده‌های طراحی شده بر فاز استنتاج [۶،۸،۹،۱۲،۱۴-۱۸،۲۶] یا فاز آموزش در انواع خاصی از الگوریتم‌های یادگیری ماشین [۱۱] تمرکز کرده‌اند.

یک شتاب‌دهنده حافظه‌ای ایده‌آل برای آموزش الگوریتم‌های یادگیری ماشین باید (۱) همه‌منظوره و قابل انعطاف باشد تا بتواند انواع الگوهای پردازشی متنوع موجود در الگوریتم‌های یادگیری ماشین را پشتیبانی کند و (۲) کارآمد باشد تا بتواند پهنای باند در دسترس حافظه‌های سه بعدی [۲۱-۲۳] را در حالی‌که محدودیت‌های توان و مساحت این حافظه‌ها را رعایت می‌کند، به دست آورد. ما مشاهده کردیم که کارهای گذشته توانایی بالقوه شتاب‌دهنده‌های درون حافظه‌ای را محدود می‌کنند. به عنوان مثال، قراردادن واحدهای همه‌منظوره [۲۷] در داخل حافظه ی پشته‌ای سه بعدی تنها تا ۱۶٪ از پهنای باند در دسترس را به دست می‌آورد (جزئیات بیشتر در بخش ۲).

ما در پی کاوش یک شتاب‌دهنده‌ی درون حافظه‌ای با واحدهای پردازشی ناهمگن برای پشتیبانی از طیف گسترده‌ای از الگوریتم‌های یادگیری ماشین هستیم. با بررسی طیف گسترده‌ای از الگوریتم‌های یادگیری ماشین، مشاهده کردیم که الگوریتم‌های یادگیری ماشین از الگوهای پردازشی مشترک استفاده می‌کنند. هر الگو می‌تواند روی یک واحد پردازشی تخصصی با سربار کم مساحت و توان اجرا شود. ترکیب این واحدهای پردازشی می‌تواند هر نوع الگوریتم یادگیری ماشین را اجرا کند. به علت محدودیت بودجه مساحت و

توان، و مشکل گرما در یک حافظه‌ی پشته‌ای سه بعدیی، تنها ٪۵۰ پهنای باند حافظه را به این واحدهای پردازشی بهینه‌شده اختصاص داده می‌شود. به منظور بهره‌برداری از تمامی پهنای باند موجود، برنامه به دو بخش برای اجرا روی واحدهای پردازشی درون حافظه (واحدهای پردازشی ناهمگن را با در نظر گرفتن بودجه مساحت و توان) و بستر پردازشی خارج از حافظه، تقسیم می شود.

PUZZLE یک راهکار سخت‌افزاری-نرم‌افزاری است که الگوهای پردازشی را به مجموعه ناهمگنی از شتاب‌دهنده‌های درون حافظه نگاشت می‌کند. PUZZLE با در نظر گرفتن موازات موجود در الگوریتم‌های یادگیری ماشین، اجرا را بر روی شتاب‌دهنده‌های درون حافظه و یک بستر پردازشی خارج از حافظه به گونه ای تقسیم می‌کند که ارتباطات بین این دو بستر پردازشی کمترین میزان ممکن باشد.

نوآوری‌های این مقاله به شرح زیر است:

• ما الگوهای پردازشی مشترک و انواع موازی‌سازی یک مجموعه از الگوریتم‌های یادگیری ماشین مختلف را استخراج می‌کنیم.

• ما PUZZLE را پیشنهاد می‌دهیم که از یک مجموعه شتاب‌دهنده‌های ناهمگن درون حافظه‌ای که از الگوهای پردازشی شناسایی‌شده استخراج شده‌اند، بهره می‌برد و پردازش را با استفاده از انواع موازی‌سازی شناسایی‌شده بر روی شتاب‌دهنده‌های درون حافظه و یک بستر پردازشی خارج از حافظه تقسیم می‌کند.

• ما نشان می‌دهیم که PUZZLE در مقایسه با بهترین راهکار موجود []، از نظر نظر کارایی و بهره‌وری انرژی به ترتیب ۱.۵ برابر و ۲۹ برابر بهتر (تا ۱.۶ برابر و ۳۱ برابر) بهتر عمل می‌کند. علاوه بر این، PUZZLE تنها در حدود ٪۱ از یک سیستم ایده‌آل، سیستمی با منابع پردازشی نامحدودی در لایه منطقی یک حافظه سه بعدی، فاصله دارد.

۲- انگیزه

دو مرحله در پردازش الگوریتم‌های یادگیری ماشین وجود دارد: (۱) مرحله آموزش که پارامترهای مدل را بر روی مجموعه داده‌های آموزشی مشخص می‌کند به گونه‌ای که خطا کاهش یابد و (۲) مرحله استنتاج که مدل آموزش دیده برای پردازش داده‌های جدید مورد استفاده قرار می‌گیرد. در حالی که هر دو مرحله از نظر پردازشی پرهزینه هستند، مرحله آموزش به دو دلیل نیازمند منابع پردازشی بیشتری است. اول اینکه، مرحله استنتاج زیرمجموعه‌ای از مرحله آموزش است. دوم اینکه، برای دستیابی به دقت بالا برای مدل‌های آموزش دیده، الگوریتم‌های یادگیری ماشین نیازمند قدرت پردازشی زیاد برای انجام پردازش بر روی مقادیر بسیار زیادی از داده‌های آموزشی هستند.

۲-۱- چالش پهنای باند حافظه

به منظور مشغول نگه‌داشتن منابع پردازشی، شتاب دهنده‌ها نیازمند انتقال مقادیر زیادی از داده‌ها هستند که این امر حافظه را به لحاظ پهنای باند و مصرف انرژی به یک گلوگاه جدی تبدیل می‌کند. حجم بالایی از کارهای پژوهشی، پردازش درون حافظه را بر پایه حافظه‌های مختلفی، مانند حافظه‌های غیرفرار و حافظه‌های سه‌بعدی به منظور بهبود عملکرد و صرفه جویی در مصرف انرژی مورد بررسی قرار داده‌اند.

در این میان مقالات پیشین، شتاب دهنده‌های درون حافظه‌ای مبتنی بر حافظه‌های سه‌بعدی، پیشنهاد کرده‌اند. حافظه‌های سه بعدی، چندین تراشه DRAM را در یک بسته بر روی هم می‌چینند. این تراشه‌ها از طریق هزاران اتصال عمودی کم‌ظرفیت از جنس سیلیکون (TSV) به یک تراشه منطقی وصل شده‌اند که کنترل‌کننده‌های حافظه در آن قرار دارند. حافظه‌های سه‌بعدی از مدارات سیگنال‌دهی پرسرعت از تراشه منطقی به تراشه فعال خارج از حافظه (مثل CPU، GPU، و FPGA) استفاده می‌کنند. به‌طور کلی حافظه‌های سه‌بعدی دارای پهنای باند بالاتر و مصرف انرژی پایین‌تر (۳ تا ۵ برابر کمتر) در مقایسه با حافظه‌های DRAM معمولی هستند.

۲-۲- چالش‌های شتاب‌دهی درون حافظه

در حالی‌که شتاب دهنده‌های درون حافظه پتانسیل رفع گلوگاه پهنای باند حافظه را دارند، دو چالش اصلی دارند که باید به آن‌ها پرداخته شود. اول اینکه، یک شتاب‌دهنده درون حافظه باید بتواند انواع مختلفی از الگوریتم‌های یادگیری ماشین را به طور مؤثر اجرا کند. دوم اینکه، محدودیت قابل توجهی در مصرف فضا و توان در حافظه‌های سه بعدی وجود دارد.

برای ارزیابی شتاب‌دهنده‌های درون حافظه، دو پارامتر تعریف می‌کنیم: (۱) عمومیت: میزان انعطاف‌پذیری معماری برای پشتیبانی از انواع مختلف الگوریتم‌های یادگیری ماشین؛ (۲) بهره‌وری: میزانی که معماری می‌تواند پهنای باند موجود را با توجه به محدودیت‌های فضا و توان به کار بگیرد.

برای دستیابی به عمومیت، یک شتاب‌دهنده درون حافظه از واحدهای اجرایی همه‌منظوره برای اجرای عملیات مختلف از جمله انواع عملیات غیرخطی در مرحله آموزش استفاده می‌کند. با این حال، واحدهای اجرایی همه‌منظوره از نظر مساحت و مصرف توان مقرون به صرفه نیستند. از سوی دیگر، برای دستیابی به بهره‌وری، نیاز است که تعداد زیادی واحد اجرایی همه‌منظوره را برای استفاده بهینه از پهنای باند، در تراشه منطقی قرار دهیم.

همان‌طور که در جدول ۱ نشان داده شده است، کارهای قبلی که از شتاب‌دهنده‌های درون حافظه [۱۱، ۱۹، ۲۰] استفاده می‌کنند، اجرای مرحله آموزش انواع مختلف الگوریتم‌های یادگیری ماشین را پشتیبانی نمی‌کنند. در حالی‌که TABLA [۲۷] یک روش کلی برای شتاب‌دهی مرحله آموزش الگوریتم‌های یادگیری ماشین است، اما از مشکل پهنای باند حافظه رنج می‌برد. سایر کارها [۲۰، ۳۳، ۴۸، ۴۹] از اجرای تقسیم‌شده بهره می‌برند اما شتاب‌دهی مرحله آموزش انواع مختلف الگوریتم‌های یادگیری ماشین را پشتیبانی نمی‌کنند. Scalpel [48]، Proger PIM [۲۰] و Resource partitioning [۳۳] فقط بخش‌های خاصی از الگوریتم‌ها را بر روی منابع مختلف اجرا می‌کنند و بقیه را فقط روی یک منبع پردازشی اجرا می‌کنند. چنین روش‌هایی نمی‌توانند موازات و متوازن‌سازی بار را فراهم کنند. راه حل‌هایی مانند تقسیم‌بندی الگوریتم‌های یادگیری ماشین بر اساس نوع لایه‌ها مانند Scaledeep [۴۹] که بخش‌های حافظه‌محور را به درون حافظه و بخش‌های پردازشی‌محور را به بستر خارج از حافظه اختصاص می‌دهند، حداقل ارتباطات بین بستری و متوازن‌سازی بار را نادیده می‌گیرند و همیشه پردازش را به صورت متوازن توزیع نمی‌کنند.

جدول ۱: شتاب‌دهنده‌های پیشین. درون حافظه (In-memory)، آموزش (Training)، عمومیت (Generality)، تقسیم‌شده (Split)، هم‌روندی (Conc)، متوازن‌سازی بار (L-B) و حداقل ارتباطات بین بستری (MinI-C).

Approach	In-memory	Training	Generality	Split	Conc	L-B	Min I-C
TABLA [27]	No	Yes	Yes	No	No	No	No
Scalpel [46]	No	Yes	No	Yes	No	No	Yes
Resource Partitioning [31]	No	No	No	Yes	No	No	No
Scaledeep [47]	No	Yes	No	Yes	No	No	No
Neurocube [11]	Yes	Yes	No	No	No	No	No
Proger PIM [20]	Yes	Yes	No	Yes	Yes	Yes	Yes
CMP-PIM [19]	Yes	No	No	No	No	No	No

از آن‌جایی که شتاب‌دهنده درون حافظه‌ای وجود ندارد که مرحله آموزش انواع مختلف الگوریتم‌های یادگیری ماشین را پشتیبانی کند، ما واحدهای اجرایی همه‌منظوره مشابهی را مانند کارهای قبلی [۲۷، ۳۲] پیاده‌سازی می‌کنیم. با در نظر گرفتن بودجه توان و فضای در دسترس حافظه‌های سه‌بعدی، این واحدها تنها ۸۰ گیگابایت بر ثانیه (۱۶٪) از پهنای باند در دسترس (از ۵۱۲ گیگابایت بر ثانیه) را به دست می‌آورند. پهنای باند به دست آمده بسیار کمتر از کل پهنای باند حافظه‌های سه‌بعدی است که نشان می‌دهد شتاب‌دهنده‌های درون حافظه همه‌منظوره نمی‌توانند پهنای باند در دسترس را به کار گیرند.

۳- PUZZLE

به منظور بهبود گلوگاه پهنای باند و شتاب‌دهی مرحله آموزش طیف گسترده‌ای از الگوریتم‌های یادگیری ماشین، ما یک رویکرد جامع درون حافظه به نام PUZZLE پیشنهاد می‌دهیم که هم نیازمندی‌های الگوریتم‌های یادگیری ماشین و هم محدودیت‌های حافظه‌های سه بعدی را برآورده می‌کند. در حالی‌که ما بر شتاب‌دهی مرحله آموزش الگوریتم‌های یادگیری ماشین تمرکز داریم، ایده پیشنهادی همچنین می‌تواند به مرحله استنتاج نیز اعمال شود، زیرا مرحله آموزش زیرمجموعه‌ای از مرحله استنتاج است.

PUZZLE از موتورهای پردازشی کم‌سربار به‌عنوان شتاب‌دهنده‌های درون حافظه سه‌بعدی بهره می‌برد تا حداکثر پهنای باند حافظه سه‌بعدی را به‌دست آورد (۲۴۰ گیگابایت بر ثانیه از ۵۱۲ گیگابایت بر ثانیه). PUZZLE از بقیه پهنای باند (۲۷۲ گیگابایت بر ثانیه) برای راه‌اندازی یک بستر پردازش خارج از حافظه استفاده می‌کند که می‌تواند ASIC، GPU، FPGA، TPU و یا هر نوع دیگری از بسترهای پردازشی باشد. PUZZLE بر اساس دو ایده اصلی اجرای آگاه از الگو و اجرای تقسیم‌شده ساخته شده است.

همان‌طور که در شکل ۱ نشان داده شده است، در ابتدا، PUZZLE به کمک یک رابط برنامه‌نویسی، روش پیشنهادی پارامترهای یادگیری (شامل نرخ یادگیری و تعداد ویژگی‌ها) و عملیات ریاضی یک الگوریتم یادگیری ماشین را دریافت می‌کند. در ادامه، گراف متناظر الگوریتم ماشین دریافتی را استخراج می‌کند (واحد ۱ در شکل ۱). سپس، PUZZLE پنج عملیات را در مرحله کامپایلر انجام می‌دهد: (۱) استخراج الگوهای پردازشی، (۲) تشخیص انواع موازی‌سازی، (۳) تقسیم گراف به دو بخش متوازن‌شده با حداقل ارتباطات بین بخشی، (۴) اختصاص هر بخش به یک بستر پردازشی، و (۵) زمان‌بندی اجرای الگوریتم بر روی دو بستر. مدیریت این اهداف در لایه کامپایلر سربار اجرایی را کاهش می‌دهد و مدیریت اجرای همزمان را تسهیل می‌کند که سازوکار کنترل را در حافظه سه‌بعدی ساده می‌کند. همان‌طور که در شکل۱ نشان داده شده است، لایه کامپایلر متشکل از دو مؤلفه اصلی است.

اجرای آگاه از الگو. در ابتدا، PUZZLE، الگوهای پردازشی در الگوریتم‌های یادگیری ماشین را شناسایی می‌کند (واحد ۲ در شکل ۱). سپس، PUZZLE هر الگوی پردازشی را با یک واحد سخت‌افزاری خاص، به نام موتور پردازشی، با سربار مساحت و توان کم پیاده‌سازی می‌کند. ترکیب این موتورهای پردازشی ناهمگن می‌تواند انواع مختلف الگوریتم‌های یادگیری ماشین را اجرا کند. استخراج‌کننده الگو سه مرحله دارد: (۱) سه زیرگراف الگو را برای سه الگوی پردازشی مشترک (reduction, comparator, optimization) ایجاد می‌کند. (۲) الگوریتم تطبیق الگو را از الگوریتم‌های گراف [۶۳، ۶۴] اقتباس و اجرا می‌کند تا تمامی نمونه‌های این الگوهای پردازشی را در گراف پیدا کند. بخش‌های باقی‌مانده گراف، نمونه‌های الگوی پردازشی خاص‌منظوره هستند. (۳) تمام گره‌ها در هر نمونه را به عنوان یک گره درشت‌دانه در گراف خوشه‌بندی می‌کند. PUZZLE مجموعه‌ای از موتورهای پردازشی ناهمگن (شکل ۲) و یک کنترل‌کننده درون حافظه را به تراشه منطقی یک حافظه سه بعدی اضافه می‌کند. کنترل‌کننده درون حافظه یک واحد سبک‌وزن است که دستورالعمل‌های بخش ۳.۱ را اجرا می‌کند.

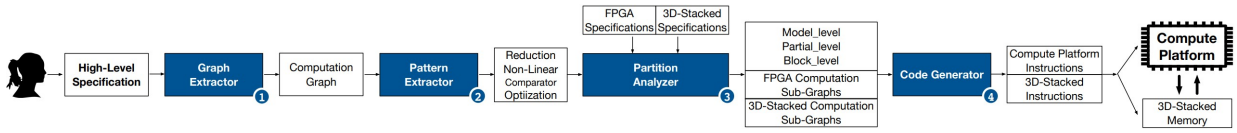
اجرای تقسیم‌شده. به منظور تقسیم بهینه یک الگوریتم یادگیری ماشین بین شتاب‌دهنده‌های درون حافظه و بستر خارج از حافظه، الگوریتم تقسیم‌بندی باید دارای سه ویژگی باشد: (۱) هم‌روندی: قسمت‌های تقسیم‌شده باید بتوانند به طور همزمان اجرا شوند. (۲) حداقل تراکنش: قسمت‌های تقسیم‌شده باید کم‌ترین ارتباطات بین‌بستری را داشته باشند. (۳) متوازن‌سازی بار: قسمت‌های تقسیم‌شده باید متناسب با قابلیت‌های پردازشی (توان پردازشی و پهنای باند حافظه موجود) دو بستر باشد. PUZZLE سه نوع موازی‌سازی را از الگوریتم‌های یادگیری ماشین استخراج می‌کند که هم‌روندی و حداقل ارتباطات بین بستری را تضمین می‌کند. سپس PUZZLE از یک الگوریتم تخصیص برای تقسیم‌بندی این بخش‌های هم‌روند بر اساس قابلیت‌های محاسباتی دو بستر استفاده می‌کند که متوازن‌سازی بار را تضمین می‌کند (واحد ۳ در شکل ۱).

۳-۱- مجموعه دستورات

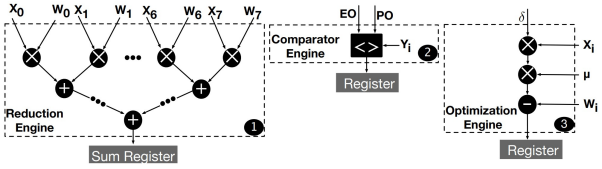
مولد کد از معماری مجموعه دستورالعمل‌های پیشنهادی ما (ISA) برای آماده‌سازی کد قابل اجرا و زمان‌بندی ایستا برای موتورهای پردازشی ناهمگن استفاده می‌کند (واحد ۴ در شکل ۱). معماری مجموعه دستورالعمل پیشنهادی ما مجموعه‌ای از دستورالعمل‌های RISC است که شامل دو پرچم، دو نوع ثبات (پردازشی و همگام‌سازی)، و سه نوع دستورالعمل (ارتباطی، پردازشی و همگام‌سازی) است. ورودی و خروجی هر موتور پردازشی به صورت سخت‌افزاری به یک ثبات پردازشی اختصاص یافته متصل است. علاوه بر ثبات‌های پردازشی ، دو ثبات برای همگام‌سازی اجرای موتورهای پردازشی و بستر خارج از حافظه وجود دارد. **دستورالعمل‌های ارتباطی** داده‌ها را از مکان‌های حافظه به ثبات‌ها و بالعکس منتقل می‌کنند (mov %src, %des).

دستورالعمل‌های پردازشی از موتورهای پردازشی برای انجام پردازش در حافظه سه بعدی استفاده می‌کنند. سه نوع دستورالعمل پردازشی وجود دارد:

۱. reduce %Num: موتور پردازشی REDUCTION با شماره Num را فعال می‌کند تا بر روی ثبات‌های ورودی خود عمل کرده و نتایج را در ثبات خروجی ذخیره کند.



شکل ۱- مراحل ایده پیشنهادی



شکل ۲- واحدهای پردازشی.

۲. comparator %Num: موتور پردازشی COMPARATOR با شماره Num را فعال می‌کند تا بر روی ثبات‌های ورودی خود عمل کرده و نتایج را در ثبات خروجی ذخیره کند.

۳. optimization %Num: موتور پردازشی OPTIMIZATION با شماره Num را فعال می‌کند تا بر روی ثبات‌های ورودی خود عمل کرده و نتایج را در ثبات خروجی ذخیره کند.

دستورالعمل‌های همگام‌سازی ارتباط مورد نیاز بین دو بستر پردازشی را مدیریت می‌کنند. در صورتی‌که واحدهای پردازشی درون و بیرون از حافظه، بخشی از یک پردازش را انجام دهند، تا زمانی که دو مجموع جزئی تجمیع نشوند، نمیتوانیم مراحل بعدی پردازش را انجام دهیم. به همین منظور، یک بستر (به عنوان ارباب شناخته می‌شود) مسئول تجمیع مجموع‌های جزئی و تولید نتیجه نهایی است، در حالی که بستر دیگر (به عنوان کارگر شناخته می‌شود) باید مجموع جزئی خود را به ارباب منتقل کند. وقتی ارباب کار خود را با مجموع جزئی تمام کرد، باید منتظر دریافت مجموع جزئی کارگر بماند. پس از دریافت مجموع جزئی، ارباب مجموع‌های جزئی را تجمیع کرده و نتیجه نهایی را برای کارگر ارسال می‌کند که منتظر نتیجه برای شروع اجرای الگوی پردازشی بهینه‌سازی است.

به همین منظور، معماری مجموعه دستورالعمل پیشنهادی از دو پرچم M_ready و S_ready، دو ثبات همگام‌سازی M_delta و S_psum و سه دستورالعمل check، set و wait استفاده می‌کند. بدون از دست دادن کلیت موضوع، فرض می‌کنیم بستر پردازشی خارج از حافظه، ارباب است. سه نوع دستورالعمل همگام‌سازی وجود دارد:

۱. set %f: مقدار پرچم %f را تنظیم می‌کند. پس از آماده‌سازی مجموع جزئی، کنترل‌کننده درون حافظه باید مجموع جزئی را در S_psum بنویسد و S_ready را با دستورالعمل set تنظیم کند. ارباب باید پرچم S_ready را بررسی کرده و هنگامی که پرچم نشان دهد آماده است، مجموع جزئی را از S_psum بخواند.

۲. wait %f: منتظر می‌ماند تا پرچم %f تنظیم شود. کنترل‌کننده درون حافظه باید منتظر بماند تا M_ready تنظیم شود و سپس مقدار ثبات M_delta را بخواند. ارباب مقدار دلتا را که برای الگوی پردازشی بهینه‌سازی مورد نیاز است محاسبه کرده، در ثبات M_delta می‌نویسد و پرچم M_ready را تنظیم می‌کند.

۳. clr %f: مقدار پرچم %f را ریست می‌کند.

۴- ارزیابی

بار کاری و مجموعه‌داده‌ها. بارهای کاری شامل الگوریتم‌های یادگیری ماشین پیشرفته هستند. الگوریتم بازپخش (BProp) مدل‌هایی را برای تشخیص ارقام دست‌نویس [۶۵، ۶۶] و گفتار [۶۷] آموزش می‌دهد. الگوریتم رگرسیون خطی (LinReg) به طور گسترده‌ای در حوزه مالی و پردازش تصویر برای پیش‌بینی قیمت [۶۸] و بافت تصاویر [۶۹] مورد استفاده قرار می‌گیرد. الگوریتم رگرسیون لجستیک (LogReg) مدل‌هایی را برای تشخیص تومورها [۷۰] و سرطان‌ها [۷۱] آموزش می‌دهد. الگوریتم ماشین بردار پشتیبان (SVM) در حوزه‌های بینایی رایانه‌ای و تشخیص پزشکی برای تشخیص چهره‌های انسانی [۷۲] و سرطان [۷۳] مورد استفاده قرار می‌گیرد. الگوریتم سیستم‌های توصیه‌گر (Reco) به طور گسترده‌ای برای پردازش مجموعه داده‌های فیلم مانند مجموعه داده‌های Movielens [۷۴، ۷۵] و مجموعه داده‌های Netflix [۷۶] مورد استفاده قرار می‌گیرد. الگوریتم رگرسیون ۲ بعدی (D-Reg۲) مدل‌هایی را برای تشخیص انواع مختلف تومورها [۷۷] و سرطان‌ها [۷۳] آموزش می‌دهد.

شبیه‌سازی. جدول ۲ پارامترهای ریزمعماری اصلی PUZZLE، مدل حافظه مورد استفاده و پارامترهای اصلی FPGA در ارزیابی‌های ما را گزارش می‌دهد. حافظه سه بعدی بر اساس یک حافظه HMC [۱۱، ۲۱، ۲۲، ۷۸] مدل‌سازی شده است. هر بخش حداکثر ۱۶ گیگابایت بر ثانیه پهنای باند به تراشه منطقی و ۱۰ گیگابایت بر ثانیه پهنای باند به تراشه فعال [۲۱، ۲۲] ارائه می‌دهد. مساحت در دسترس برای شتاب‌دهنده‌ها در هر بخش ۱.۵ میلی‌متر مربع است [۶، ۲۱]. ما پارامترهای مدل حافظه سه بعدی را از برگه داده‌ها [۲۱] استخراج کرده‌ایم. ما سخت‌افزار را در پلتفرم FPGA با استفاده از Vivado Design Suite v2017.2 سنتز می‌کنیم تا پارامترهای طراحی FPGA را استخراج کنیم. ما از Synopsys Design Compiler (L-2016.03-SP5) و کتابخانه سلول استاندارد TSMC 45-nm با فرکانس ۳۱۳ مگاهرتز (فرکانس حافظه سه‌بعدیHMC) [۱۱، ۲۱، ۲۲، ۷۸]، برای سنتز شتاب‌دهنده و به دست آوردن اعداد مربوط به مساحت، تأخیر و انرژی استفاده می‌کنیم. ما از CACTI-P [۷۹] برای اندازه‌گیری مساحت و توان ثبات‌ها و SRAMهای درون تراشه استفاده می‌کنیم. با استفاده از اعداد حاصل از سنتز و پیکربندی مدل‌های حافظه، ما یک شبیه‌ساز سطح چرخه پیاده‌سازی کرده‌ایم تا عملکرد و مصرف انرژی PUZZLE را اندازه‌گیری کنیم. شبیه‌ساز PUZZLE

شامل زمان‌بندی دسترسی به حافظه است و به طور دقیق پارامترهای پیاده‌سازی‌های ASIC و FPGA را مدل می‌کند.

معیارهای مقایسه. ما منافع PUZZLE را با شش الگوریتم یادگیری ماشین، از نظر عملکرد و حاصل‌ضرب توان–تأخیر (EDP) ارزیابی می‌کنیم.

جدول ۲–پارامترهای طراحی

3D-Stacked Memory	Model	HMC v 2.1
	Capacity	8 GB
	Number of Vaults	32
	Number of Banks/Vault	16
	Number of Links/Package	4
	Logic Die Frequncy	313 MHz
	Internal Access Latency	27.5 ns [22]
	Total Bandwidth	512 GB/s
	Internal Transfer Energy	3.7 pJ/bit [22]
	External Access Latency	27.5 ns [22]
	External Transfer Energy	10 pJ/bit [22]
	Area per vault (mm2)	1.5

FPGA	Model	UltraScale+ VU13P
	Technology Node	16 nm
	Peak Frequency	250 MHz
	Total Number of LUTs	1,728 K
	Total Number of Flip-Flops	3,456 K
	Total Number of DSPs	12,288
	BRAM Size	94.5 Mb
	UltraRAM Size	360 Mb

PUZZLE	# of Reduction Units per Logic Die	8
	# of Multipliers per Reduction Unit	8
	# of Adders per Reduction Unit	7
	# of Optimization Units	64
	# of Multipliers per Optimization Unit	2
	Total Area (mm2)	41.3
	Maximum Captured Bandwidth (GB/s)	240

بسترهای مقایسه. ما شش بستر مختلف را مقایسه می‌کنیم.

PUZZLE رویکرد ماست که در آن هم از FPGA و هم از موتورهای پردازشی بر روی تراشه منطقی حافظه سه بعدی استفاده می‌کنیم. جدول ۵ همچنین منابع موجود بر روی تراشه منطقی که PUZZLE برای محاسبه از آن استفاده می‌کند را فهرست می‌کند.

PUZZLE-IIC یک PUZZLE با ارتباط ایده‌آل بین بستری با تأخیر صفر و بدون استفاده از پهنای باند را ارزیابی می‌کند.

برای طرح FPGA، ما واحدهای محاسبه و منطق [۲۷] را در یک FPGA متصل به حافظه سه‌بعدی پیاده‌سازی می‌کنیم.

PIM-GU، از آنجا که شتاب‌دهنده درون حافظه‌ای وجود ندارد که مرحله آموزش انواع مختلف الگوریتم‌های یادگیری ماشین را پشتیبانی کند، این طرح از واحدهای همه‌منظوره مشابه کارهای قبلی [۲۷، ۳۲] بر روی تراشه منطقی حافظه استفاده می‌کند.

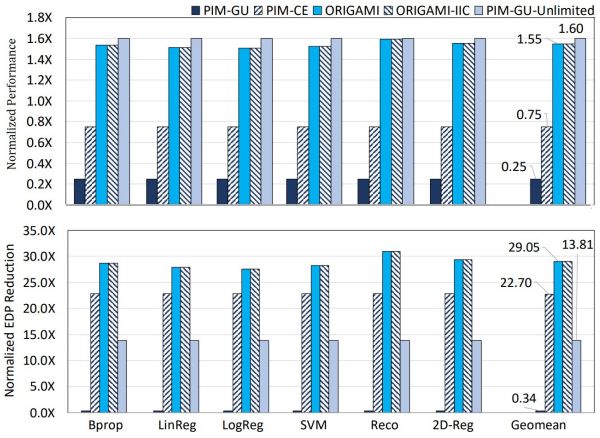
PIM-GU-Unlimited، یک بستر ایده‌آل است که از تمام پهنای باند موجود استفاده می‌کند.

PIM-CE یک PUZZLE را ارزیابی می‌کند که فقط از موتورهای پردازشی بر روی تراشه منطقی بهره می‌برد.

جدول ۶ مشخصات شتاب‌دهنده‌های PUZZLE و واحد محاسبه و منطق کارهای قبلی [۲۷، ۳۲] را نشان می‌دهد. ما فقط می‌توانیم ۳۲ واحد محاسبه و منطق را بر روی تراشه منطقی قرار دهیم، زیرا مساحت یک واحد محاسبه و منطق ۱.۲ میلی‌متر مربع است و مساحت در دسترس در یک بخش ۱.۵ میلی‌متر مربع می‌باشد. در نتیجه، PIM-GU از ۸۰ گیگابایت بر ثانیه از پهنای باند در دسترس استفاده می‌کند.

۵- نتایج ارزیابی

تحلیل کارایی. شکل ۳، زمان اجرا را در تمام بارهای کاری نشان می‌دهد. نتایج نسبت به FPGA نرما شده‌اند. ما چهار مشاهده اصلی داریم. اول اینکه، PUZZLE در مقایسه با FPGA از نظر زمان اجرا به طور میانگین ۱.۵۵ برابر (حداکثر ۱.۶ برابر) بهتر عمل می‌کند. PUZZLE از تمام پهنای باند در دسترس استفاده می‌کند. دوم اینکه، شتاب PUZZLE در حاشیه‌ی ۱٪ از PIM-GU-Unlimited قرار دارد. PUZZLE با توزیع درست پردازش بین FPGA و شتاب‌دهنده‌ها بر روی تراشه منطقی حافظه سه بعدی، به طور بهینه‌ای از پهنای باند حافظه استفاده می‌کند. سوم اینکه، کارایی PUZZLE با PUZZLE-IIC برابر است که نشان‌دهنده کارایی بالای الگوریتم تقسیم‌بندی ما در کمینه‌کردن ارتباطات بین بستری است. ارزیابی‌های ما نشان می‌دهد که سربار پهنای باند در PUZZLE کمتر از ۰.۰۰۱٪ است و PUZZLE به طور مؤثری سربار تأخیر را پنهان می‌کند. چهارم اینکه، PIM-CE در مقایسه با PIM-GU ۲.۹ برابر سریع‌تر است که نشان‌دهنده کارایی موتورهای پردازشی ناهمگن در مقایسه با واحدهای همه‌منظوره است. علاوه بر این، PUZZLE موتورهای پردازشی ناهمگن و اجرای تقسیم‌شده را ترکیب می‌کند تا تمام پهنای باند در دسترس را به دست آورد و در نتیجه در مقایسه با PIM-CE ۱.۲ برابر سریع‌تر عمل می‌کند.



شکل ۳– ارزیابی کارایی و بهره‌وری انرژی.

تحلیل انرژی. شکل ۳ کاهش حاصل‌ضرب توان–تأخیر (EDP) را نشان می‌دهد که نسبت به FPGA نرمال شده‌اند. ما چهار مشاهده اصلی داریم. اول اینکه EDP در PUZZLE در مقایسه با FPGA به طور میانگین ۲۹ برابر (حداکثر ۳۱ برابر) کمتر است. دلیل آن دو چیز است: (۱) در PUZZLE، بخشی از ارتباطات داده درون حافظه سه بعدی است و (۲) PUZZLE از موتورهای پردازشی سبک استفاده

می‌کند که نسبت به واحدهای پردازشی همه‌منظوره FPGA مصرف انرژی کمتری دارند. دوم اینکه، EDP در PUZZLE به طور میانگین به ترتیب ۸۶.۱ برابر و ۲.۱ برابر کمتر از PIM-GU و PIM-GU-Unlimited است. همه واحدهای پردازشی در PIM-GU و PIM-GU-Unlimited درون حافظه قرار دارند ولی این واحدها نسبت به موتورهای پردازشی PUZZLE مصرف انرژی بیشتری دارند. سوم اینکه، PIM-CE به طور میانگین به ترتیب ۶۷.۳ برابر و ۱.۷ برابر بهتر از PIM-GU و PIM-GU-Unlimited عمل می‌کند. دلیل آن استفاده PIM-CE از موتورهای پردازشی ناهمگن است که مصرف انرژی آن‌ها به طور قابل توجهی کمتر از واحدهای پردازشی همه‌منظوره موجود در PIM-GU و PIM-GU-Unlimited است. چهارم اینکه، PUZZLE و PUZZLE-IIC به دلیل ارتباطات کم بین‌بستری ارائه‌شده توسط اجرای تقسیم‌شده PUZZLE تقریباً EDP یکسانی دارند.

۶- پژوهش‌های پیشین

PUZZLE با کارهای قبلی از دو جهت متفاوت است: (۱) استخراج الگوهای پردازشی در الگوریتم‌های یادگیری ماشین و نگاشت آن‌ها به موتورهای پردازشی ناهمگن بر روی تراشه منطقی یک حافظه سه بعدی، (۲) تقسیم اجرای الگوریتم‌های یادگیری ماشین بر روی موتورهای پردازشی درون حافظه و یک بستر پردازشی خارج از حافظه، به صورت متوازن‌شده و با حداقل ارتباطات بین بستری.

تعداد زیادی معماری برای شتاب‌دهنده‌های درون حافظه وجود دارد که واحدهای پردازشی را بر روی یک تراشه یک‌پارچه می‌کنند تا پهنای باند حافظه بالاتر و مصرف انرژی دسترسی پایین‌تر را فراهم کنند [۱، ۲، ۵–۷، ۹–۱۴، ۱۶–۲۰، ۲۷–۳۱، ۳۴–۴۷، ۸۰]. اکثر این معماری‌های درون حافظه مرحله استنتاج الگوریتم‌های یادگیری ماشین را تسریع می‌بخشند. برخی از شتاب‌دهنده‌های درون حافظه مانند Neurocube [۱۱] و Proger PIM [۲۰] هر دو مرحله آموزش و استنتاج را شتاب می‌دهند، اما فقط برای شبکه‌های عصبی پیچیده کار می‌کنند و برای سایر الگوریتم‌های یادگیری ماشین کاربرد ندارند.

کارهای گذشته از بسترهای ASIC [۱، ۲، ۵، ۳۸–۴۷]، GPU [۳۴–۳۷]، FPGA [۲۷–۳۱] و چندگره‌ی پردازشی [۴۷، ۸۰] برای تسریع الگوریتم‌های یادگیری ماشین استفاده کرده‌اند. در حالی‌که این روش‌ها مؤثر هستند، اما از پردازش درون حافظه بهره‌مند نمی‌شوند. کارهای گذشته از اجرای تقسیم شده برای تسریع الگوریتم‌های یادگیری ماشین استفاده کرده‌اند. Shen و همکاران [۳۳] منابع FPGA را برای پردازش زیرمجموعه‌های مختلف لایه‌های کانولوشن شبکه‌های عصبی کانولوشنی تقسیم‌بندی کردند. Scalpel [۴۸] ساده‌سازی شبکه را با استفاده از روش‌های هرس وزن مبتنی بر آگاهی از SIMD و هرس گره انجام داده‌اند. Park و همکاران [۳۲] فقط بخش بهینه‌سازی مرحله آموزش الگوریتم‌های مختلف یادگیری ماشین را بر روی FPGA و n واحد ASIC توزیع می‌کنند. در نتیجه، روش آن‌ها متوازن‌سازی بار را ارائه نمی‌دهد. Scaledeep [۴۹] از کاشی‌های پردازشی ناهمگن استفاده می‌کند که برای بخش‌های پردازشی‌محور و حافظه‌محور آموزش شبکه‌های عصبی ژرف سفارشی‌سازی شده‌اند. Proger PIM [۲۰] از CPU، PIM ثابت و یک واحد PIM برنامه‌پذیر برای آموزش مدل‌های مختلف شبکه عصبی کانولوشنی استفاده می‌کند. این روش‌ها از نظر بنیادی با روش پیشنهادی ما متفاوت هستند زیرا هیچ‌کدام پردازش درون حافظه نیستند و ویژگی‌های لازم برای پردازش درون حافظه کارا را ارائه نمی‌دهند.

۷- نتیجه گیری

در طول مرحله آموزش، الگوریتم‌های یادگیری ماشین مقادیر زیادی از داده‌ها را پردازش می‌کنند که مصرف پهنای باند و انرژی قابل توجهی دارد. اگرچه شتاب‌دهنده‌های درون حافظه، پهنای باند حافظه بالا و مصرف انرژی کمتری فراهم می‌کنند، اما از نظر عمومیت یا بهره‌وری دچار مشکل هستند. ما PUZZLE را پیشنهاد می‌کنیم، یک رویکرد جامع که از موتورهای پردازشی ناهمگن بر روی تراشه منطقی بهره می‌برد تا به طور مؤثر طیف گسترده‌ای از الگوریتم‌های یادگیری ماشین را پوشش دهد و اجرای الگوریتم‌های یادگیری ماشین را بر روی موتورهای پردازشی درون حافظه و یک بستر پردازشی خارج از حافظه تقسیم می‌کند تا از تمام پهنای باند موجود استفاده کند. نتایج ارزیابی نشان می‌دهد که PUZZLE در مقایسه با بهترین کار قبلی، از نظر کارایی و حاصل‌ضرب توان–تأخیر (EDP) به ترتیب تا ۱.۶ برابر و ۳۱ برابر بهتر عمل می‌کند. PUZZLE همچنین به طور میانگین کارایی و بهره‌وری انرژی را به ترتیب ۱.۵ برابر و ۲۱ برابر بهبود می‌بخشد.

مراجع

[1] Y.-H. Chen, J. Emer, and V. Sze, “Eyeriss: A spatial architecture for energy-efficient dataflow for convolutional neural networks,” ISCA, 2016.
[2] Z. Du, R. Fasthuber, T. Chen, P. Ienne, L. Li, T. Luo, X. Feng, Y. Chen, and O. Temam, “Shidiannao: Shifting vision processing closer to the sensor,” ISCA, 2015.
[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” NeurIPS, 2012.
[4] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” 2014.
[5] J. Albericio, P. Judd, T. Hetherington, T. Aamodt, N. E. Jerger, and A. Moshovos, “Cnvlutin: Ineffectual-neuron-free deep neural network computing,” ISCA, 2016.
[6] M. Gao, J. Pu, X. Yang, M. Horowitz, and C. Kozyrakis, “Tetris: Scalable and efficient neural network acceleration with 3d memory,” ASPLOS, 2017.
[7] L. Song, X. Qian, H. Li, and Y. Chen, “Pipelayer: A pipelined reram-based accelerator for deep learning,” HPCA, 2017.

A unified template-based framework for accelerating statistical machine learning," HPCA, 2016.

[28] C. Zhang, P. Li, G. Sun, Y. Guan, B. Xiao, and J. Cong, "Optimizing FPGA-based Accelerator Design for Deep Convolutional Neural Networks," in Intl. Symp. FPGA, 2015.

[29] H. Sharma, J. Park, D. Mahajan, E. Amaro, J. K. Kim, C. Shao, A. Misra, and H. Esmaeilzadeh, "From high-level deep neural models to fpgas," MICRO, 2016.

[30] A. R. Putnam, D. Bennett, E. Dellinger, J. Mason, and P. Sundararajan, "Chimps: A high-level compilation flow for hybrid cpu-fpga architectures," FPGA, 2008.

[31] C. Farabet, Y. LeCun, K. Kavukcuoglu, E. Culurciello, B. Martini, P. Akselrod, and S. Talay, "Large-scale FPGA-based convolutional networks," Scaling up Machine Learning: Parallel and Distributed Approaches, 2011.

[32] J. Park, H. Sharma, D. Mahajan, J. K. Kim, P. Olds, and H. Esmaeilzadeh, "Scale-out acceleration for machine learning," MICRO, 2017.

[33] Y. Shen, M. Ferdman, and P. Milder, "Maximizing cnn accelerator efficiency through resource partitioning," ISCA, 2017.

[34] S. G. Elango, Convolutional Neural Network Acceleration on GPU by Exploiting Data Reuse. PhD thesis, San Jose State University, 2017.

[35] K.-S. Oh and K. Jung, "GPU implementation of neural networks," Pattern Recognition, 2004.

[36] A. Guzhva, S. Dolenko, and I. Persiantsev, "Multifold acceleration of neural network computations using GPU," ICANN, 2009.

[37] K. Li, J. Chen, W. Chen, and J. Zhu, "Saberlda: Sparsity-aware Learning of Topic Models on GPUs," ASPLOS, 2017.

[38] H. Sharma, J. Park, N. Suda, L. Lai, B. Chau, J. Kyung Kim, V. Chandra, and H. Esmaeilzadeh, "Bit fusion: Bit-level dynamically composable architecture for accelerating deep neural networks," ISCA, 2018.

[39] V. Aklaghi, A. Yazdanbakhsh, K. Samadi, H. Esmaeilzadeh, and R. K. Gupte, "Snapea: Predictive early activation for reducing computation in deep convolutional neural networks," ISCA, 2018.

[40] A. Yazdanbakhsh, K. Samadi, H. Esmaeilzadeh, and N. S. Kim, "GANAX: A Unified SIMD-MIMD Acceleration for Generative Adversarial Network," ISCA, 2018.

[41] P. Judd, J. Albericio, T. Hetherington, T. M. Aamodt, and A. Moshovos, "Stripes: Bit-serial Deep Neural Network Computing," MICRO, 2016.

[42] A. Parashar, M. Rhu, A. Mukkara, A. Puglielli, R. Venkatesan, B. Khailany, J. Emer, S. W. Keckler, and W. J. Dally, "SCNN: An Accelerator for Compressed-sparse Convolutional Neural Networks," ISCA, 2017.

[43] S. Han, X. Liu, H. Mao, J. Pu, A. Pedram, M. A. Horowitz, and W. J. Dally, "EIE: Efficient Inference Engine on Compressed Deep Neural 12 Network," ISCA, 2016.

[44] S. Zhang, Z. Du, L. Zhang, H. Lan, S. Liu, L. Li, Q. Guo, T. Chen, and Y. Chen, "Cambricon-X: An Accelerator for Sparse Neural Networks," MICRO, 2016.

[45] Y. Chen, T. Luo, S. Liu, S. Zhang, L. He, J. Wang, L. Li, T. Chen, Z. Xu, N. Sun, et al., "Dadiannao: A machine-learning supercomputer," MICRO, 2014.

[46] D. Liu, T. Chen, S. Liu, J. Zhou, S. Zhou, O. Teman, X. Feng, X. Zhou, and Y. Chen, "Pudiannao: A polyvalent machine learning accelerator," ASPLOS, 2015.

[47] N. P. Jouppi, C. Young, N. Patil, D. Patterson, G. Agrawal, R. Bajwa, S. Bates, S. Bhatia, N. Boden, A.

[8] L. Nai, R. Hadidi, J. Sim, H. Kim, P. Kumar, and H. Kim, "GraphPIM: Enabling Instruction-Level PIM Offloading in Graph Computing Frameworks," HPCA, 2017.

[9] K. Hsieh, E. Ebrahimi, G. Kim, N. Chatterjee, M. O'Connor, N. Vijaykumar, O. Mutlu, and S. W. Keckler, "Transparent Offloading and Mapping (TOM): Enabling Programmer-Transparent Near-Data Processing in GPU Systems," ISCA, 2016.

[10] A. Shafiee, A. Nag, N. Muralimanohar, R. Balasubramonian, J. P. Strachan, M. Hu, R. S. Williams, and V. Srikumar, "Isaac: A convolutional neural network accelerator with in-situ analog arithmetic in crossbars," 2016.

[11] D. Kim, J. Kung, S. Chai, S. Yalamanchili, and S. Mukhopadhyay, "Neurocube: A Programmable Digital Neuromorphic Architecture with High-Density 3D Memory," ISCA, 2016.

[12] H. Asghari-Moghaddam, Y. Hoon Son, J. Ho Ahn, and N. Sung Kim, "Chameleon: Versatile and Practical Near-DRAM Acceleration Architecture for Large Memory Systems," MICRO, 2016.

[13] P. Chi, S. Li, C. Xu, T. Zhang, J. Zhao, Y. Liu, Y. Wang, and Y. Xie, "Prime: a novel processing-in-memory architecture for neural network computation in reram-based main memory," ISCA, 2016.

[14] A. Farmahini-Farahani, J. H. Ahn, K. Morrow, and N. S. Kim, "NDA: Near-DRAM Acceleration Architecture Leveraging Commodity DRAM Devices and Standard Memory Modules," HPCA, 2015.

[15] J. Ahn, S. Hong, S. Yoo, O. Mutlu, and K. Choi, "A Scalable Processing-in-Memory Accelerator for Parallel Graph Processing," ISCA, 2015.

[16] A. Farmahini-Farahani, J. H. Ahn, K. Morrow, and N. S. Kim, "DRAMA: An Architecture for Accelerated Processing Near Memory," CAL, 2015.

[17] Q. Zhu, T. Graf, H. Sumbul, L. Pileggi, and F. Franchetti, "Accelerating Sparse Matrix-Matrix Multiplication with 3D-Stacked Logic-in-Memory Hardware," HPEC, 2013.

[18] C. Farabet, B. Martini, B. Corda, P. Akselrod, E. Culurciello, and Y. LeCun, "Neuflow: A runtime reconfigurable dataflow processor for vision," CVPRW, 2011.

[19] S. Angizi, Z. He, A. S. Rakin, and D. Fan, "Cmp-pim: an energy-efficient comparator-based processing-in-memory neural network accelerator," DAC, 2018.

[20] J. Liu, H. Zhao, M. A. Ogleari, D. Li, and J. Zhao, "Processing-in-memory for energy-efficient neural network training: A heterogeneous approach," MICRO, 2018.

[21] Hybrid Memory Cube Consortium, Hybrid Memory Cube Specification 2.1, 6 2014. Rev. 10.0.

[22] "Hybrid memory cube."

[23] P. Rosenfeld, Performance Exploration of the Hybrid Memory Cube. PhD thesis, 2014.

[24] T. Zhang, K. Wang, Y. Feng, Y. Chen, Q. Li, B. Shao, J. Xie, X. Song, L. Duan, Y. Xie, et al., "A 3d soc design for h. 264 application with on-chip dram stacking," 3DIC, 2010.

[25] M. Ghosh and H.-H. S. Lee, "Smart refresh: An enhanced memory controller design for reducing energy in conventional and 3D die-stacked DRAMs," MICRO, 2007.

[26] L. Zhou, S. Pan, J. Wang, and A. V. Vasilakos, "Machine learning on big data: Opportunities and challenges," 2017.

[27] D. Mahajan, J. Park, E. Amaro, H. Sharma, A. Yazdanbakhsh, J. K. Kim, and H. Esmaeilzadeh, "Tabla:

[70] M. Segal, K. Dahlquist, and B. Conklin, "Regression approaches for microarray data analysis," Journal of Computational Biology, 2003.
[71] D. Singh, P. Febbo, K. Ross, D. Jackson, J. Manola, C. Ladd, P. Tamayo, A. Renshaw, A. A. D, J. Richie, E. Lander, M. Loda, P. Kantoff, T. Golub, and W. Sellers, "Gene expression correlates of clinical prostate cancer behavior," Cancer Cell, 2002.
[72] J. Weston, S. Mukherjee, O. Chapelle, M. Pontil, T. Poggiott, and V. Vapnik, "Feature selection for svms," NIPS, 2000.
[73] "Integrated cancer repository for cancer research."
[74] I. Cantador, P. Brusilovsky, and T. Kuflik, "Movielens dataset," HetRec, 2011.
[75] Grouplens, "Movielens dataset," 2017.
[76] "Netflix prize data set."
[77] "Integrated cancer repository for cancer research."
[78] J. Jeddeloh and B. Keeth, "Hybrid Memory Cube New DRAM Architecture Increases Density and Performance," VLSIT, 2012.
[79] S. Li, K. Chen, J. H. Ahn, J. B. Brockman, and N. P. Jouppi, "CACTI-P: Architecture-level Modeling for SRAM-based Structures with Advanced Leakage Reduction Techniques," ICCAD, 2011.
[80] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mane, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viegas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous distributed systems," arXiv, 2016.
Borchers, et al., "In-datacenter performance analysis of a tensor processing unit," ISCA, 2017.
[48] J. Yu, A. Lukefahr, D. Palframan, G. Dasika, R. Das, and S. Mahlke, "Scalpel: Customizing dnn pruning to the underlying hardware parallelism," ISCA, 2017.
[49] S. Venkataramani, A. Ranjan, S. Banerjee, D. Das, S. Avancha, A. Jagannathan, A. Durg, D. Nagaraj, B. Kaul, P. Dubey, et al., "Scaledeep: A scalable compute architecture for learning and evaluating deep networks," ISCA, 2017.
[50] J. Zhang, Z. Wang, and N. Verma, "A machine-learning classifier implemented in a standard 6t sram array," VLSI-Circuits, 2016.
[51] J. Ahn, S. Yoo, O. Mutlu, and K. Choi, "Pim-enabled instructions: A low-overhead, locality-aware processing-in-memory architecture," ISCA, 2015.
[52] C. De Sa, M. Feldman, C. Ré, and K. Olukotun, "Understanding and optimizing asynchronous low-precision stochastic gradient descent," ISCA, 2017.
[53] L. Bottou, "Stochastic gradient learning in neural networks," Neuro-Nımes, 1991.
[54] L. Bottou, "Stochastic gradient descent tricks," in Neural networks: Tricks of the trade, 2012.
[55] S. Zhang, C. Zhang, Z. You, R. Zheng, and B. Xu, "Asynchronous stochastic gradient descent for dnn training," ICASSP, 2013.
[56] R. Ormándi, I. H. us1, and M. Jelasity, "Asynchronous peer-to-peer data mining with stochastic gradient descent," Euro-Par, 2011.
[57] F. Seide, H. Fu, J. Droppo, G. Li, and D. Yu, "1-bit stochastic gradient descent and its application to data-parallel distributed training of speech dnns," Interspeech, 2014.
[58] M. Li, T. Zhang, Y. Chen, and A. J. Smola, "Efficient mini-batch training for stochastic optimization," KDD, 2014.
[59] J. Kaufmann, "Signal conditioner with symbol addressed lookup table producing values which compensate linear and non-linear distortion using transversal filter," 1998. US Patent 5,778,029.
[60] K. Engel, M. Kraus, and T. Ertl, "High-quality pre-integrated volume rendering using hardware-accelerated pixel shading," HWWS, 2001.
[61] T. A. Keahey and E. L. Robertson, "Techniques for non-linear magnification transformations," ISIV, 1996.
[62] J. Mielikainen et al., "Lossless compression of hyperspectral images using lookup tables," Signal Process, 2006.
[63] C. Schulz, "Graph partitioning and graph clustering in theory and practice," KIT, 2016.
[64] G. W. Flake, R. E. Tarjan, and K. Tsioutsiouliklis, "Graph clustering and minimum cut trees," Internet Mathematics, 2004.
[65] Y. LeCun and C. Cortes, "MNIST handwritten digit database," 2010.
[66] "A variant of mnist dataset with 8 millions records."
[67] J. P. Pinto, Multilayer Perceptron Based Hierarchical Acoustic Modeling for Automatic Speech Recognition. PhD thesis, EPFL, 2010.
[68] B. Zhou, "High-frequency data and volatility in foreign-exchange rates," Journal of Business & Economic Statistics, 2008.
[69] S. Dhanya and R. V. Kumari, "Comparison of various texture classification methods using multiresolution analysis and linear regression modelling," Springerplus, 2016.

پنجمین کنفرانس ملی انفورماتیک ایران
پژوهشگاه دانشهای بنیادی، پردیس فرمانیه، تهران
۱۳ و ۱۴ دی ماه ۱۴۰۲

پردازش بهینه مرحله آموزش شبکه عصبی با استفاده از اشتراک داده

زهرا رحیمی[۱]، هاجر فلاحتی[۲]، حاکم بیت الهی[۳]

[۱] دانشجو، دانشکده مهندسی کامپیوتر، دانشگاه علم و صنعت ایران، تهران،
BaharRahimi1500@gmail.com
[۲] استادیار، استادیار پژوهشکده کامپیوتر، پژوهشگاه دانش‌های بنیادی، تهران،
hfalahati@ipm.ir
[۳] استادیار، دانشکده مهندسی کامپیوتر، دانشگاه علم و صنعت ایران، تهران،
beitollahi@iust.ac.ir

چکیده

امروزه با بزرگ‌تر شدن و پیچیده‌تر شدن شبکه‌های عصبی، پردازش آن‌ها در مرحله استنتاج و آموزش به منابع پردازشی و حافظه بیشتری نیاز دارد. با افزایش هزینه انتقال داده و اهمیت حفظ امنیت داده‌ها، اجرای مرحله آموزش شبکه‌های عصبی در دستگاه‌های لبه مورد توجه قرار گرفته است. مرحله آموزش، شامل عملیات انتشار به جلو، انتشار به عقب و به روز رسانی وزن است و نیاز به حافظه و پردازش بیشتری نسبت به استنتاج دارد، در نتیجه با توجه به محدودیت منابع و انرژی در دستگاه‌های لبه، اجرای آن با چالش‌های بیشتری رو به رو است.

در این پژوهش ما یک رویکرد فشرده‌سازی آگاه از داده برای بهره‌برداری از پراکندگی و شباهت در بردارهای ورودی، وزن و گرادیان در مرحله آموزش برای شتاب‌دهنده‌های شبکه‌های عصبی ژرف پیشنهاد می‌کنیم و یک معماری شتاب‌دهنده جدید جهت اجرای جریان داده، برای کاهش بی نظمی‌های ایجاد شده پیشنهاد می‌کنیم. بررسی عملکرد روش پیشنهادی بر روی سه شبکه عصبی ژرف نشان داده است که روش پیشنهادی در مقایسه با شتاب‌دهنده آگاه از تکرار و پراکندگی و یک روش آگاه از شباهت داده ورودی از لحاظ کارایی به ترتیب ۴.۷× و ۶.۸× و از لحاظ کاهش انرژی مصرفی به ترتیب ۳.۵× و ۷.۲× بهبود داشته است.

کلمات کلیدی

شبکه‌های عصبی ژرف، شتاب‌دهنده سخت‌افزاری، آموزش، فشرده‌سازی، پراکندگی.

۱- مقدمه

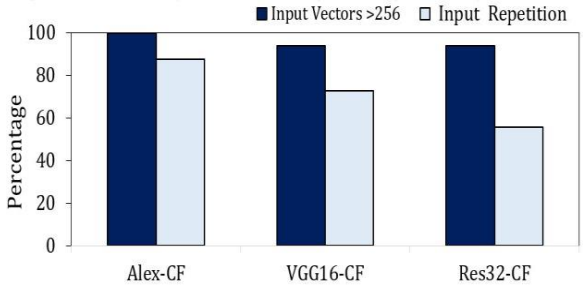
امروزه شبکه‌های عصبی ژرف کاربردهای بسیاری در زمینه‌های مختلف مانند پردازش تصویر، کاربردهای پزشکی، وسایل نقلیه خودران و دوربین‌های نظارتی دارند [1, 2]. هرچند حجم حافظه و پردازش بالای مورد نیاز در اجرای شبکه‌های عصبی ژرف، چالش اساسی برای اجرای این شبکه‌ها بر روی دستگاه‌های لبه (با محدودیت‌های شدید انرژی و منابع) به‌شمار می‌رود [3]. یکی از بسترهای پردازشی مناسب برای اجرای شبکه‌های عصبی، مخصوصا مرحله آموزش آن‌ها، سرورهای ابری هستند. سرورهای ابری توان پردازشی و حافظه مناسبی دارند ولی با چالش‌هایی مانند هزینه‌های انرژی و تأخیر بالا برای انتقال داده‌ها بین ابر و دستگاه و محرمانه بودن داده‌ها مواجه هستند. در سال‌های پیشین، اجرای آموزش در دستگاه‌های لبه به عنوان یکی از راه‌حل‌های مورد توجه برای جلوگیری از چنین مشکلاتی مطرح شده است [4].

آموزش شبکه‌های عصبی ژرف به دلیل پردازش انتشار به عقب و عملیات اضافی مورد نیاز برای به روز رسانی پارامترها، نیاز پردازشی و حافظه بیشتری نسبت به مرحله استنتاج دارد. از سوی دیگر، برای دستیابی به دقت مورد نظر، مرحله آموزش بر روی حجم بالایی از داده‌ها تکرار می‌شود. به عنوان نمونه، در هر بار آموزش شبکه عصبی، نیاز به ذخیره‌سازی داده‌های تولید شده در هر لایه و به روز رسانی پارامترهای مختلف داریم [5].

تلاش‌های زیادی برای کاهش پیچیدگی مدل‌های شبکه‌های عصبی ژرف و تسریع آموزش آن‌ها انجام شده است. روش‌های فشرده‌سازی، آموزش توزیع شده، استفاده از پراکندگی و شباهت داده‌ها از جمله‌ی این کارها هستند که با کاهش حجم پردازش، اجرای برنامه را در بسترهای لبه امکان‌پذیر می‌کنند [6]. نتایج نشان داده است که پتانسیل بالایی در استفاده همزمان از پراکندگی و شباهت در پارامترهای شبکه عصبی وجود دارد. شتاب‌دهنده‌های آموزشی پیشین [7-12] به بررسی پراکندگی در پارامترهای مختلف شبکه عصبی پرداخته‌اند و [6, 13-15] از استفاده دوباره ورودی‌ها در مرحله آموزش بهره‌برداری کرده‌اند. شتاب‌دهنده‌های [16, 17] از اشتراک داده و پراکندگی در مرحله استنتاج استفاده کرده‌اند. با این حال، بررسی شباهت در وزن و ورودی و گرادیان همراه با پراکندگی هنوز در مرحله آموزش مورد بهره‌برداری قرار نگرفته است. به‌کارگیری همزمان هر دو روش می‌تواند نیاز به پردازش و حافظه را نسبت به شتاب‌دهنده‌های قبلی در مرحله آموزش شبکه‌های عصبی کاهش دهد.

بررسی ما نشان داد که امکان اشتراک داده در مرحله آموزش بسیار بیشتر از مرحله استنتاج است. چرا که آموزش شامل دو مرحله انتشار به جلو و

انتشار به عقب می‌باشد. شکل (۱) درصد تکرار داده‌ها در بردارهای ورودی سه شبکه عصبی مختلف را نشان می‌دهد. با تجزیه و تحلیل بردارهای ورودی با استفاده از داده‌های ۸ بیتی (پژوهش‌های پیشین [18] نشان داده‌اند که می‌توان در مرحله آموزش از داده‌های ۸ بیتی استفاده کرد و تنها در زمان به روز رسانی وزن‌ها، از پردازش ۳۲ بیتی استفاده می‌شود) مشاهده می‌کنیم درصد زیادی از ورودی‌ها بیشتر از ۲۵۶ تا هستند که بیانگر وجود تکرار در داده‌های ورودی است. این تکرارها را برای مدل‌های AlexNet، VGG-16 و ResNet32 به ترتیب ۸۷.۵، ۷۳ و ۵۶ درصدگزارش می‌کنیم. بنابراین با بهره‌برداری از اشتراک داده‌ها در عملیات‌های شبکه عصبی، می‌توان در کاهش میزان پردازش داده‌ها و حجم حافظه شاهد بهبود چشمگیری بود.



شکل (۱) : درصد تکرار داده‌ها در بردارهای ورودی سه شبکه عصبی AlexNet، VGG16 و ResNet32

در این پژوهش، ما یک راهکار نرم‌افزاری- سخت‌افزاری ارائه کرده‌ایم که به صورت همزمان از فرصت‌های پراکندگی و اشتراک داده در وزن‌ها، ورودی‌ها و گرادیان‌ها در پردازش کانولوشن شبکه‌های عصبی استفاده می‌کند. اولین مرحله از رویکرد پیشنهادی، تشخیص وزن‌های یکتا در هر ورودی از لایه‌ها و مشخص کردن آدرس ورودی‌های متناظر با آن وزن‌ها است. مرحله دوم به این صورت انجام می‌شود که با آمدن ورودی‌های هر لایه، مقادیر یکتای غیرصفر آن‌ها تشخیص داده می‌شود و در وزن‌های یکتای غیرصفر متناظر با آن‌ها ضرب می‌شود. در مرحله سوم، هر ورودی یکتای غیرصفر تنها به یک وزن یکتا اختصاص داده می‌شود. برای حفظ درستی عملیات مورد نظر، نیاز به استفاده از فراداده [1]داریم که در ادامه، پردازش و سربار آن‌ها را به طور کامل بیان می‌کنیم.

ارزیابی معماری پیشنهادی بر روی سه شبکه عصبی ژرف محبوب نشان می‌دهد که رویکرد پیشنهادی ما از نظر کارایی به ترتیب ×۴.۷، ×۶.۸ و از نظر مصرف انرژی به ترتیب ×۳.۵، ×۷.۲ نسبت به شتاب‌دهنده پیشرفته آگاه از تکرار و پراکندگی(UCNN) و یک روش آگاه از شباهت داده ورودی (ADR) بهبود دارد. نوآوری‌های این پژوهش عبارتند از:

- ما مشاهده کردیم که شباهت زیادی در بین بردارهای عصبی در مرحله انتشار به جلو و انتشار به عقب در مرحله آموزش شبکه عصبی ژرف وجود دارد.
- ما یک معماری شتاب‌دهنده پویا ارائه کرده‌ایم که مختص به یک شبکه عصبی ژرف خاص نمی‌باشد و به بهره‌برداری همزمان از پراکندگی و تکرار داده‌ها در مرحله آموزش شبکه عصبی با تغییر ترتیب پردازش‌ها و طرح داده‌ها به صورت موازی می‌پردازد.
- ارزیابی جامع سه شبکه عصبی ژرف که بر روی مجموعه داده CIFAR-10 با استفاده از داده‌های ۸ بیتی آموزش داده شده‌اند، نشان داده است که شتاب‌دهنده پیشنهادی عملکرد بهتری نسبت به شتاب‌دهنده‌های پیشرفته پراکندگی/تکرار دارد.

۲- پیشینه

شبکه عصبی دارای دو مرحله پردازشی است: استنتاج و آموزش. در مرحله استنتاج، داده‌های ورودی به ترتیب از لایه‌ها عبور می‌کنند و هر لایه ضرب‌های ماتریس را روی داده‌ها انجام می‌دهد [19]. آموزش شبکه‌های عصبی ژرف که بر اساس الگوریتم گرادیان نزول[2] است شامل دو مرحله انتشار به جلو و انتشار به عقب است. مرحله انتشار به جلو شامل پردازش مرحله استنتاج شبکه عصبی است. در آموزش از خروجی آخرین لایه در مرحله انتشار به جلو برای یافتن خطای پیش‌بینی (گرادیان‌ها) نسبت به خروجی‌های صحیح استفاده می‌شود. سپس گرادیان‌ها به ترتیب معکوس از آخرین لایه به لایه اول برای به روز رسانی پارامترهای شبکه منتشر می‌شوند. در حین انتشار به جلو، خروجی لایه i بر اساس فرمول زیر محاسبه می‌شود.

$$A[i+1] = A[i] \times W[i] \quad (۱)$$

در جایی که برای لایه i، A[i] فعال‌سازی ورودی است و W[i] ماتریس وزن است. خروجی لایه i به عنوان ورودی لایه بعدی (i + 1) استفاده می‌شود. برای انتشار خطای پیش‌بینی در هر لایه i، از گرادیان ورودی و گرادیان وزن استفاده می‌کنیم که در رابطه (۲) و (۳) نشان داده شده‌اند.

$$G[i-1] = G[i] \times (W[i])^T \quad (۲)$$

$$G_W[i] = A[i] \times G[i] \quad (۳)$$

G[i-1] گرادیان فعال‌سازی ورودی، G[i] گرادیان فعال‌سازی خروجی و $G_W[i]$ گرادیان وزن است. T یک عملیات چرخش عمومی را معرفی می‌کند که به موجب آن ماتریس W به اندازه ۱۸۰درجه چرخش داده می‌شود. گرادیان وزن در یک دسته برای به روز رسانی وزن‌ها برای یک گذر بر اساس رابطه (۴) استفاده می‌شود که در آن W[i+1] وزن جدید، W[i] وزن قدیمی، $G_W[i]$ گرادیان وزن و α نرخ یادگیری است[6].

$$W[i+1] = W[i] - \alpha \times G_W[i] \quad (۴)$$

۳- بهره‌برداری از تکرار داده‌ها

امروزه نیاز به تسریع بیشتر آموزش و بهبود بهره‌وری انرژی در کاربردهای لبه بیش از پیش است [20]. برای شتاب‌دهنده شبکه‌های عصبی اگر سخت‌افزار از دقت کامل (ممیز شناور ۳۲ بیتی) برای پشتیبانی از آموزش استفاده کند، مساحت و سربار انرژی بسیار زیاد خواهد بود. به همین دلیل شبیه‌سازی‌های نرم‌افزاری برای انجام آموزش کم‌بیت با کوانتیزه‌کردن داده‌ها انجام شده است [4]. ما در این پژوهش از داده‌های با دقت پایین‌تر یعنی نقطه ثابت ۸

[1]Meta data

[2] Gradient descent

بیتی برای آموزش استفاده می‌کنیم که با وجود استفاده از داده‌های با عرض بیت کم، به روز رسانی وزن‌ها همچنان به پردازش داده‌های با دقت بالا (ممیز شناور ۳۲ بیتی) نیاز دارد [18]. به همین دلیل، ما علی‌رغم استفاده از داده‌های ۸ بیتی برای پردازش مرحله انتشار به جلو و انتشار به عقب، برای پردازش به روز رسانی وزن‌ها از داده‌های ممیز شناور ۳۲ بیتی استفاده می‌کنیم.

کوانتیزیشن داده‌ها در شبکه‌های عصبی، با کاهش عرض بیت داده‌ها باعث ایجاد داده‌های مشابه می‌شود. بنابراین فاکتورگیری از داده‌های مشابه به ما فرصت کاهش عملیات ضرب وجمع در پردازش و کاهش حجم حافظه را می‌دهد. با این حال، استفاده از این روش‌ها در طول آموزش چالش برانگیزتر از مرحله استنتاج است. به همین دلیل شتاب‌دهنده‌های موجود که استنتاج را پشتیبانی می‌کنند برای آموزش کافی نیستند [10].

برای بهره‌برداری از پتانسیل موجود در داده‌های اشتراکی، در این پژوهش، بر روی عملیات کانولوشن تمرکز کردیم و قصد داریم داده‌های تکراری را شناسایی و با فاکتورگیری آن‌ها، تنها یک بار هر داده را ذخیره و پردازش کنیم. رابطه (۵) فرآیند فاکتورگیری و استخراج شباهت در روش پیشنهادی ما را به صورت مرحله به مرحله نشان می‌دهد. این رابطه، عملیات‌های کانولوشن موجود در شبکه، شامل تولید خروجی هر لایه در مرحله انتشار به جلو، رابطه (۱)، گرادیان وزن، رابطه (۲)، و گرادیان ورودی، رابطه (۳)، می‌باشد.

مرحله ۱:

$$OF[z] = \sum_{i=0}^{Nw[z]} Dof_{Nu}[i] \times \sum_{j=0}^{NI[z][i]} Don[j]$$

مرحله ۲:

$$= \sum_{i=0}^{Nw[z]} Dof_{Nu}[i] \times \left(\sum_{l=0}^{NI[z][i]} Don_{Nu}[l] \times Count[z][l][i]\right)$$

مرحله ۳:

$$= \sum_{i=0}^{Nw[z]} Dof_{Nu}[i] \times \left(\sum_{u=0}^{Nu[z][i]} Don_{Nu}[u] \times \boldsymbol{Mov}[z][u][i]\right)$$

$Don_{Nu}[u]$ is a non-zero Unique element assigned to the $Dof_{Nu}[i]$

$$\mathbf{Mov}[z][u][i] = \sum_{s=0}^{Ns[z]} Count[z][u][s] \times (-1)^{\mathbf{sign}} \times \mathbf{Shift[s][i]}$$

$$\mathbf{Shift[s][i]} = Dof_{Nu}[s] / Dof_{Nu}[i]$$

$$\mathbf{Sign} = \begin{cases} 0, & \text{if both source and destination } Dof_{Nu} \text{ are positive or negative} \\ 1, & \text{Otherwise} \end{cases} \quad (۵)$$

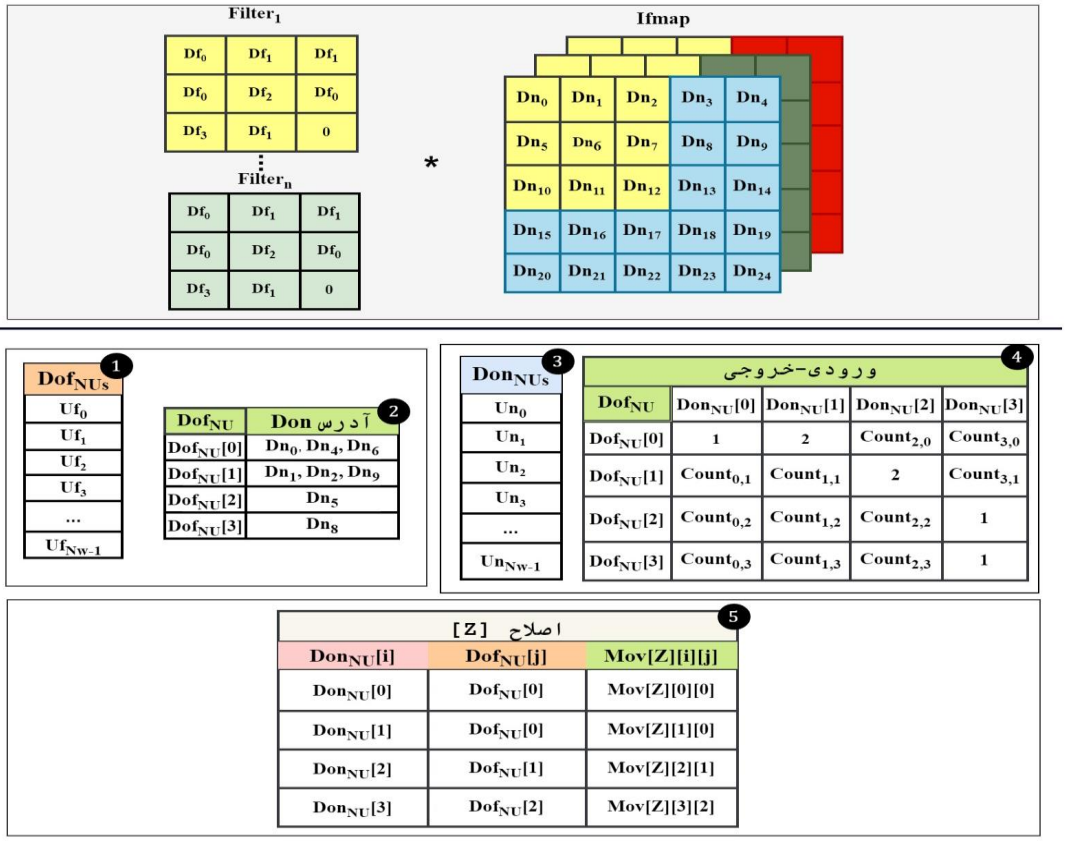
در این رابطه متغیر Don خلاصه شده Data Online می‌باشد و به داده‌هایی که در طول پردازش شبکه عصبی به صورت برخط تولید می‌شوند (خروجی تولید شده هر لایه در مرحله انتشار به جلو، گرادیان وزن و گرادیان ورودی در مرحله انتشار به عقب) اطلاق می‌گردد. به همین صورت، Dof خلاصه شده Data Offline می‌باشد و به داده‌هایی که اطلاعات آن از قبل در شبکه موجود است (وزن‌ها در مرحله انتشار به جلو، وزن‌ها و فعال‌سازی‌ها در مرحله انتشار به عقب) اطلاق می‌گردد. به همین صورت Don_{Nu} و Dof_{Nu} مقادیر یکتای غیرصفر این داده‌ها هستند که غالباً در فاکتورگیری به کار برده می‌شوند. عملیات شکل (۲) مراحل انجام شده از اجرای روش ما برای تولید یک نمونه خروجی در یک لایه از شبکه عصبی را نشان می‌دهد.

عملیات کانولوشن در مرحله انتشار به جلو به صورت رابطه (۱) است. همان‌طور که می‌دانیم، در این مرحله وزن‌ها به صورت ایستا در شبکه موجود می‌باشند (Dof). در این روش، برای همه فیلترها در هر لایه وزن‌های یکتای غیرصفر (Dof_{Nu}) را شناسایی کرده❶ و آدرس ورودی‌های متناظر با آن‌ها را استخراج می‌کنیم❷. سپس از عملیات فاکتورگیری وزن‌های یکتای تکراری استفاده می‌کنیم (مرحله۱). در حین انتشار به جلو، با دریافت فعال‌سازی هر لایه ورودی‌های یکتای غیرصفر (Don_{Nu}) را شناسایی می‌کنیم❸ و پنجره‌های بعدی را به همین صورت پردازش می‌کنیم. در نهایت، برای هر خروجی، تعداد دفعاتی که هر ورودی یکتای غیرصفر با هر وزن یکتای غیرصفر ضرب شده‌است را می‌شماریم (Count[z][l][i])❹. با این کار تعداد تکرارهای هر ورودی یکتای غیرصفر برای هر وزن یکتای غیرصفر مشخص می‌شود (مرحله۲).

در مرحله انتشار به عقب ما با دو عملیات کانولوشن دیگر رو به رو هستیم که رابطه آن‌ها به ترتیب در رابطه (۲) و (۳) آمده است. همان‌طور که می‌دانیم، داده‌های میانی تولید شده از مرحله انتشار به جلو برای برگشت به عقب استفاده می‌شود. بنابراین، یک سری از اطلاعات برای محاسبه مرحله انتشار به عقب را می‌توان از قبل استخراج کرد. برای محاسبه گرادیان ورودی و گرادیان وزن، چون اطلاعات وزن‌ها و فعال‌سازی‌ها و مقادیر یکتای غیرصفر آن‌ها را از مرحله قبل داریم پس می‌توان در صورت تغییر (چرخش ۱۸۰ درجه برای وزن‌ها در گذر به عقب)، آن‌ها را حساب کرد و آدرس گرادیان متناظر هر مقدار یکتا را استخراج کرد. در واقع با دریافت خروجی لایه قبل و اعمال تابع ضرر[3] روی آن، گرادیان‌های خروجی را محاسبه می‌کنیم. سپس گرادیان‌های یکتای غیرصفر را یک بار برای همه فیلترها در هر لایه شناسایی کرده و پنجره‌های بعدی را هم به همین صورت پردازش می‌کنیم. در نهایت، گرادیان‌های یکتای غیرصفر و تعداد تکرارهای آن را برای هر وزن یا فعال‌سازی یکتای غیرصفر می‌شماریم. با این کار گرادیان‌های مشابه را در هر عبارت فاکتوری یک خروجی فاکتور می‌گیریم.

فاکتورگیری داده‌های یکتا با در نظر گرفتن حداقل تعداد عناصر ورودی، دستورالعمل‌های پردازشی را به طور میانگین به میزان ۱۴٪ کاهش می‌دهد و موجب کاهش دسترسی به داده‌ها در حافظه هم خواهد شد. اما علی‌رغم مزیت آن جهت کاهش پردازش و کاهش حجم حافظه با چالشی رو به رو است. این‌که یک داده برخط یکتای غیرصفر ممکن است به چند عبارت فاکتوری تعلق داشته باشد که موجب سربار ذخیره‌سازی زیادی می‌شود. به منظور کاهش این هزینه اضافی، در رویکرد پیشنهادی هر داده برخط یکتای غیرصفر را فقط یک بار در هر عبارت فاکتور در نظر می‌گیریم و آن را به عبارت وزنی اولی که مربوط به آن است منتقل می‌کنیم. برای مثال در عملیات انتشار به

[3] loss



شکل (۲): یک مثال از اجرای یک لایه در شتاب‌دهنده پیشنهادی

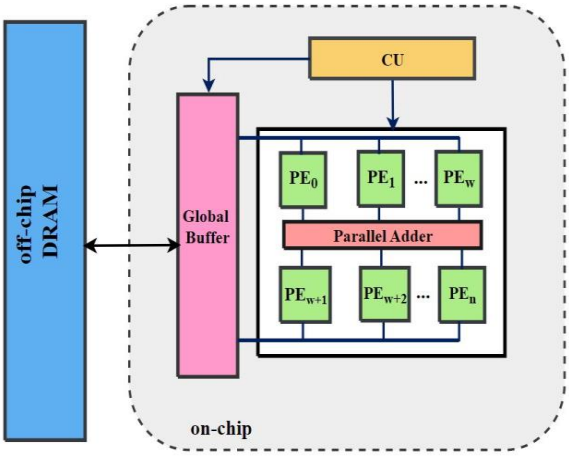
جلو، هر ورودی یکتا غیر صفر را در تمام وزن ها به ترتیب وزن‌ها بررسی کرده و در صورتی که ورودی موردنظر در دیگر وزن‌ها هم باشد آن را به اولین عبارت وزنی که در آن وجود داشته انتقال می‌دهیم. با انجام این کار، داده‌های یکتای غیرصفر برای همه‌ی پارامترهای شبکه عصبی به تعداد یک مرتبه پردازش می‌شود که در نتیجه عملیات جمع و ضرب را کاهش می‌دهد، اما این کار ترتیب پردازش را از بین می‌برد که موجب بی نظمی قابل توجهی در شبکه می‌شود. برای حل این مشکل، پارامتر Mov را محاسبه می‌کنیم (مرحله۳) که روش محاسبه آن در رابطه (۵) نشان داده شده است.

Mov یک پارامتر است که تعیین می‌کند داده‌های غیرصفر چگونه از یک عبارت فاکتور به عبارت فاکتور دیگر منتقل شوند. این پارامتر تعداد داده‌های برخط یکتای غیرصفر را که روی وزن مبدأ قرار دارند، در مقدار علامت و شیفت ضرب می‌کند. در واقع Mov[z][u][i] یک ابرداده است که نشان می‌دهد چگونه داده برخط یکتای غیرصفر $Don_{Nu}[u]$ از عبارت فاکتور مبدأ DofNu[s] به عبارت فاکتور مقصد DofNu[i] جابه‌جا شود. علامت (sign) نیز برای مشخص کردن علامت داده‌ها استفاده می‌شود. برای کاهش سربار ذخیره‌سازی، مقادیر داده‌های یکتای غیرصفر را به صورت توان دو در نظر می‌گیریم. به همین دلیل در Mov برای انتقال داده برخط یکتا از داده غیر برخط یکتای مبدأ به مقصد به شیفت نیاز داریم که روش محاسبه آن در رابطه (۵) نشان داده شده است. با استفاده از پارامتر Mov، پردازش ثابت می‌ماند و فقط یک جفت اطلاعات ایندکس برای هر داده برخط یکتای غیرصفر ارسال می‌شود. به این ترتیب شتاب‌دهنده عملیات‌های بسیاری را برای همه‌ی داده‌های موجود در هر لایه از شبکه عصبی اصلاح می‌کند❺ و عملیات اصلاح شده را به صورت موازی اجرا می‌کند.

۳-۱- معماری

شتاب‌دهنده کلی از چندین عنصر پردازش[4](PEs)، یک واحد کنترل[5](CU) و یک بافر جهانی[6] همان‌طور که در شکل (۳) نشان داده شده است، تشکیل شده است. هر PE شامل واحدهایی برای انجام محاسبات است و داده‌های خود را توسط گذرگاه‌هایی دریافت می‌کند. هر PE، از یک ضرب‌کننده موازی، جمع کننده موازی، ضرب‌کننده و سه ثبات ۸ بیتی ساخته شده است که برای ذخیره مجموع داده‌های برخط یکتای غیرصفر مربوطه، داده‌های غیر برخط یکتای غیرصفر و مجموع جزئی یک عبارت فاکتور اختصاص داده شده‌اند. هنگامی که داده‌ها در بافرجهانی در دسترس هستند، واحد کنترل داده‌ها را از بافر جهانی می‌خواند و داده‌های غیر برخط یکتای موجود را یک بار در هر لایه به PEها اختصاص می‌دهد. بنابراین شتاب‌دهنده به اندازه تعداد داده‌های غیر برخط یکتا، PE خواهد داشت. هر PE یک عبارت فاکتور را محاسبه می‌کند و حاصل را در ثبات مجموع جزئی ذخیره می‌کند. شتاب‌دهنده مجموع جزئی PEها را به وسیله یک جمع‌کننده موازی که در توپولوژی PEها قرار گرفته شده است جمع می‌کند و یک عنصر خروجی را محاسبه می‌کند. در نهایت، خروجی محاسبه شده را در بافر جهانی ذخیره می‌کند تا به عنوان ورودی برای لایه بعدی استفاده شود.

[4] Processing Element
[5] Control Unit
[6] Global Buffer



شکل (۳) : شتاب‌دهنده پیشنهادی

۴- روش شناسی

روش پیشنهادی، به کمک شبیه‌سازی با زبان برنامه نویسی پایتون با استفاده از کتابخانه‌های TensorFlow و keras انجام شده‌است. جدول (۱) مجموعه‌ای از شبکه‌های عصبی ژرف مورد استفاده در این پژوهش، ساختار و دقت آن‌ها را برحسب فاکتورهای مورد استفاده در برنامه‌های کاربردی آن‌ها فهرست می‌کند. به عنوان مثال، Top-1، Top-5. دقت یک شبکه به عنوان تعداد داده‌های طبقه‌بندی شده به طور صحیح به تعداد کل عناصر در یک مجموعه داده آزمایشی تعریف می‌شود. ما شبکه‌های عصبی ژرف مطرح شده را با مقادیر نقطه ثابت ۸ بیتی در مجموعه داده CIFAR-10 [21] (تشخیص شی آموزش دیده) آموزش می‌دهیم. برای آموزش، ما از بهینه‌ساز SGD با مومنتوم ۰.۹ و نرخ یادگیری اولیه ۰.۱ برای ۲۰۰ دوره روی یک GPU RTX 2080 Ti با اندازه دسته‌ای ۱۲۸ در ۳۹۰ تکرار استفاده می‌کنیم و میزان دقت آن‌ها را ثبت می‌کنیم.

برای استخراج ویژگی‌های طراحی، مدل‌های RTL شتاب‌دهنده و همه طرح‌های مورد قیاس با آن را توسعه می‌دهیم. برای مقایسه منصفانه، همه طرح‌های مقایسه شده دارای پیکربندی یکسانی هستند که در فرکانس ۱ گیگاهرتز کار می‌کنند و به یک حافظه DRAM با پهنای باند ۲۵۶ گیگابایت بر ثانیه و انرژی ۱۲.۵ pJ/bit در هر دسترسی متصل هستند. ما PEها را با استفاده از Design Compiler با تکنولوژی ۲۸ نانومتر سنتز می‌کنیم. شبیه‌ساز سطح بالای ما مراحل پیش‌پردازش را در سمت نرم‌افزار انجام می‌دهد که روی یک پردازنده Core-i7 با فرکانس ۴ گیگاهرتز و حافظه SSD، ۵۱۲ گیگابایتی کار می‌کند

جدول (۱) : شبکه‌های عصبی ژرف مورد استفاده در پژوهش

نام مدل	مجموعه داده	تعداد لایه	پارامترهای دقت			
			۸ بیت		۳۲ بیت	
			Top-1	Top-5	Top-1	Top-5
AlexNet	CIFAR10	۸	۷۱.۰	۹۷.۴	۷۱.۴	۹۷.۶
VGG16	CIFAR10	۱۶	۸۸.۴	۹۹.۵	۸۸.۷	۹۹.۵
ResNet32	CIFAR10	۳۴	۸۸.۸	۹۹.۵	۸۶.۷	۹۹.۶

۴-۱- تجزیه و تحلیل

پهنای بیت: در این پژوهش ما یک نمایش نقطه ثابت ۸ بیتی را برای آموزش در نظر می‌گیریم. در فرآیند فاکتورگیری قبل از هر ضرب جمع‌هایی بین داده‌ها انجام می‌شود. برای مثال در عملیات کانولوشن مرحله انتشار به جلو، عملوند ورودی از عملوند وزن گسترده‌تر خواهد بود. بدترین حالت زمانی است که اندازه گروه فعال‌سازی به اندازه کل پنجره ورودی باشد. به این معنی که کل پنجره با یک وزن یکتای غیر صفر مطابقت داشته باشد. جهت کاهش هزینه‌های ایجاد شده برای اندازه گروه فعال‌سازی حداکثر محدودیت ۴k را تعیین می‌کنیم. بنابراین در این حالت تعداد تکرار ورودی ۱۲ بیت می‌شود. پارامتر Mov، ۸ بیتی است زیرا مجموع چند عبارت ضرب (Shift و Count) است. در اینجا ما فرض می‌کنیم به طور متوسط، هر عنصر ورودی یکتای غیرصفر ۱۶ بار در هر عبارت وزنی تکرار می‌شود که این تعداد (Count) توسط ۴ بیت کدگذاری می‌شود. از آنجایی که تعداد (Count) در بدترین حالت ممکن است ۱۲ بیت باشد، در این حالت بیت‌های اضافی را برای هر مقدار در نظر می‌گیریم و آن‌ها را جداگانه انتقال می‌دهیم. همچنین پارامتر Shift توسط حداکثر ۸ عملیات پردازش می‌شود که توسط ۳ بیت کدگذاری می‌شود.

زمان: برای تجزیه و تحلیل زمان، تعداد عملیات در هر چرخه را با استفاده از ویژگی‌های سخت‌افزاری مانند PEهای فعال و میزان استفاده از آن‌ها در نظر می‌گیریم. رویکرد پیشنهادی طبق رابطه (۵)، مرحله ۱ را پردازش می‌کند و مراحل دیگر شامل ارسال داده‌ها و اطلاعات ایندکس به سخت‌افزار و پردازش خروجی در سمت سخت‌افزار را خط لوله می‌کند. ما تمام این مراحل پردازش و اصلاح در سمت سخت‌افزار را در شبیه‌ساز اجرا می‌کنیم و تعداد چرخه‌های لازم برای آن‌ها را تعیین می‌کنیم. مرحله ۱ که به صورت غیربرخط اجرا می‌شود، یک پنجره را با ۱۶۴ چرخه برای تمام خروجی‌های مربوطه پردازش می‌کند. سپس مرحله ۲ و مرحله ۳ که به صورت برخط اجرا می‌شوند برای هر خروجی مربوطه فعال می‌شوند. مرحله ۲ ایندکس داده‌های مربوط برای پارامترهای Mov را با ۵ چرخه استخراج می‌کند و مرحله ۳ داده را با ۳۰ چرخه ارسال می‌کند. واحد کنترل، داده‌ها را در یک چرخه به هر PE ارسال می‌کند. هر PE عملیات جمع را در یک چرخه و عملیات ضرب را در یک چرخه (زمانی که عملوند توان دو است) یا ۲ چرخه انجام می‌دهد. در مجموع هر PE، ۳–۵ چرخه را برای محاسبه یک عبارت فاکتور صرف می‌کند. جمع کننده موازی نتایج تمام PEها را در یک چرخه جمع می‌کند. در نتیجه، شتاب‌دهنده یک خروجی را در ۵–۷ چرخه محاسبه می‌کند. ارسال داده‌ها به شتاب‌دهنده ۳۰ چرخه طول می‌کشد. بنابراین، شتاب‌دهنده ۶ خروجی را پردازش می‌کند.

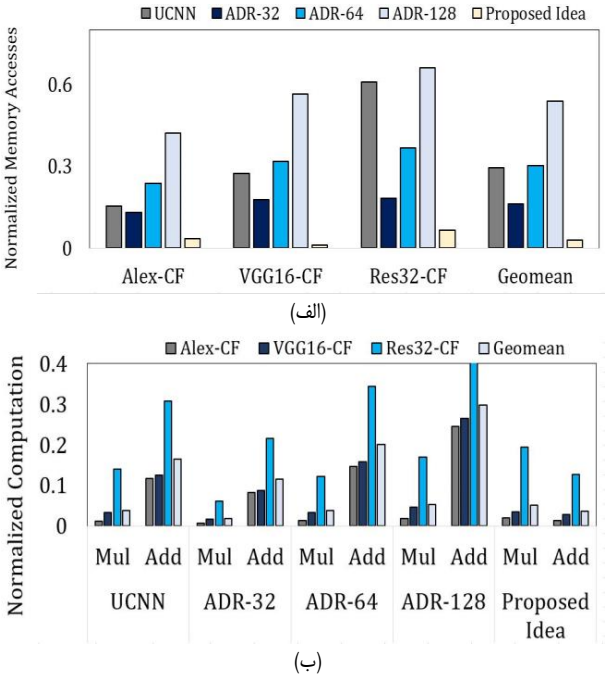
انرژی: برای تجزیه و تحلیل انرژی، انرژی در هر عملیات را اندازه‌گیری کرده و آن را در تعداد عملیات ضرب می‌کنیم. برای محاسبه انرژی، فعالیت‌های واحد را از طریق شبیه‌سازی در Modelsim در نظر می‌گیریم.

۵- ارزیابی

ما شتاب‌دهنده پیشنهادی خود را با یک شتاب‌دهنده شبیه Eyeriss [22] (شتاب‌دهنده DNN پایه)، UCNN [16] (شتاب‌دهنده پیشرفته آگاه از

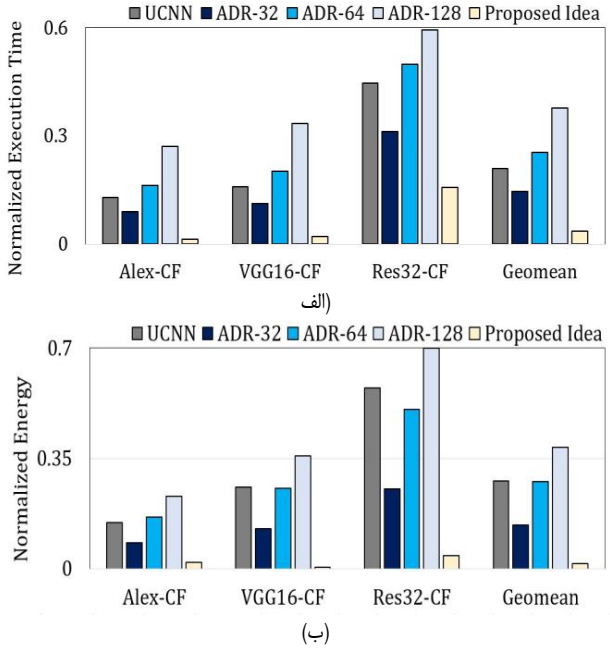
تکرار و پراکندگی) و ADR [13] (یک روش آگاه از شباهت داده ورودی) با اندازه‌های دسته‌ای ۳۲، ۶۴، ۱۲۸ مقایسه می‌کنیم.

تعداد دسترسی‌های حافظه / دستورالعمل‌های پردازشی: دراین پژوهش یکی از اهداف اصلی که به دنبال آن بوده‌ایم بحث کاهش حجم حافظه اصلی و کاهش تعداد عملیات‌های پردازشی است که از جمله چالش‌های اصلی آموزش شبکه‌های عصبی بر روی دستگاه های لبه با محدودیت منابع می‌باشد. شکل (۴) تعداد دسترسی‌های حافظه و دستورالعمل‌های پردازشی را به صورت نرمال شده نسبت به طرح Eyeriss نشان می‌دهد. مشاهده می‌کنیم که شتاب‌دهنده پیشنهادی در مقایسه با طرح‌های Eyeriss، UCNN، ADR-32، ADR-64، ADR-128 (۱) دسترسی‌های حافظه را به میزان ×۳۴.۴، ×۳.۳۸، ×۶.۱، ×۳.۳، ×۱.۸۵ و (۲) تعداد ضرب‌ها را به ترتیب ×۱۹.۶، ×۲۵.۹، ×۵۲.۳، ×۲۶.۱، ×۱۸.۷ و تعداد جمع‌ها را به ترتیب ×۲۷.۴، ×۶، ×۸.۶، ×۴.۹، ×۳.۳ کاهش داده است. دلیل چنین برتری این است که شتاب‌دهنده پیشنهادی ۱) هم پراکندگی و هم شباهت را در وزن‌ها، ورودی‌ها و گرادیان‌ها در نظر می‌گیرد، در حالی که UCNN فقط پراکندگی و تشابه در وزن‌ها و ADR فقط تشابه در ورودی‌ها را در نظر می‌گیرند. ۲) فقط یک بار عناصر یکتا را پردازش می‌کند.



شکل (٤) : مقایسه شتاب‌دهنده پیشنهادی نسبت به طرح‌های Eyeriss، UCNN، ADR-32، ADR-64، ADR-128: الف) دسترسی به حافظه ب) تعداد عملیات‌های پردازشی

کارایی/ مصرف انرژی: ملاک کارایی این طرح کاهش زمان پردازش آموزش نسبت به طرح‌های موجود برای شبکه‌های عصبی ژرف مختلف می‌باشد. شکل (۵) نمودار زمان پردازش و انرژی مصرفی را به صورت نرمال شده نسبت به طرح Eyeriss نشان می‌دهد. شتاب‌دهنده پیشنهادی نسبت به طرح‌های Eyeriss، UCNN، ADR-32، ADR-64، ADR-128، به ترتیب (۱) ×۲۷.۱، ×۴.۷، ×۶.۸، ×۳.۹، ×۲.۶ زمان پردازش و (۲) ×۵۹.۸، ×۳.۵، ×۷.۲، ×۳.۶، ×۲.۵ مصرف انرژی کم‌تری دارد. با توجه به بهبودهای حاصل، طرح ما در کاربردهای لبه که نیاز به مصرف پایین انرژی دارند به خوبی عمل خواهد کرد.



شکل (٥) : مقایسه شتاب‌دهنده پیشنهادی نسبت به طرح‌های Eyeriss، UCNN، ADR-32، ADR-64، ADR-128: الف) زمان اجرا ب) انرژی مصرفی

٦- نتیجه‌گیری

نیاز به تسریع بیشتر مرحله آموزش شبکه‌های عصبی در بسترهای پردازشی نهفته، به دلیل کمبود منابع پردازشی و حافظه و کاربردهایی چون مراکز داده و لبه نمود پیدا کرده است. برای رسیدگی به این چالش‌ها برخی از شتاب‌دهنده‌های پیشین از قابلیت‌های فشرده‌سازی شبکه‌های عصبی برای تسریع روند آموزش و کارایی انرژی آن پرداختند. با این حال، آن‌ها نتوانستند از هر دو فرصت پراکندگی و شباهت برای آموزش شبکه‌های عصبی به طور همزمان استفاده کنند. در این پژوهش ما یک طرح نرم‌افزاری - سخت‌افزاری با سربار کم را پیشنهاد می‌کنیم که با بهره‌برداری از فرصت‌های پراکندگی و شباهت همه پارامترهای شبکه عصبی به صورت همزمان، به حل چالش‌های آموزشی در کاربردهای لبه می‌پردازد. ما شتاب‌دهنده‌ای را پیشنهاد کردیم که به طور موثر از تکرار داده‌های ورودی، وزن و گرادیان در پردازش کانولوشن استفاده می‌کند، به بهره‌برداری از داده‌های یکتای غیر صفر به تعداد یک مرتبه می‌پردازد و عملیات کانولوشن را در لحظه اصلاح می‌کند. طبق ارزیابی‌های انجام شده، شتاب‌دهنده‌ی پیشنهادی از نظر کارایی و انرژی بهبودهای بیشتری را نسبت به روش‌های پیشین شاهد است.

مراجع

[1] S. Huai, L. Zhang, D. Liu, W. Liu, and R. Subramaniam, *"Zerobn: Learning compact neural networks for latency-critical edge systems,"* in 2021 58th ACM/IEEE Design Automation Conference (DAC), 2021: IEEE, pp. 151-156 .

[2] N. Louloudakis, P. Gibson, J .Cano, and A. Rajan, *"DeltaNN: Assessing the Impact of Computational Environment Parameters on the Performance of Image Recognition Models,"* in 2023 IEEE International Conference on Software Maintenance and Evolution (ICSME), 2023: IEEE, pp. 414-424 .

[3] S. Voghoei, N. H. Tonekaboni, J. G. Wallace, and H. R. Arabnia, *"Deep learning at the edge,"* in 2018 International Conference on Computational Science and Computational Intelligence (CSCI), 2018: IEEE, pp. 895-901 .

[4] S. Choi, J. Shin, Y. Choi, and L.-S .Kim, *"An optimized design technique of low-bit neural network training for personalization on IoT devices,"* in Proceedings of the 56th Annual Design Automation Conference 2019, 2019, pp. 1-6 .

[5] Y. Sepehri, P. Pad, A. C. Yüzügüler, P. Frossard, and L .A. Dunbar, *"Hierarchical Training of Deep Neural Networks Using Early Exiting,"* arXiv preprint arXiv:2303.02384, 2023.

[6] J. Servais and E. Atoofian, *"Adaptive computation reuse for energy-efficient training of deep neural networks,"* ACM Transactions on Embedded Computing Systems (TECS), vol. 20, no. 6, pp. 1-24, 2021.

[7] X. Sun, X. Ren, S. Ma, and H. Wang, *"meprop: Sparsified back propagation for accelerated deep learning with reduced overfitting,"* in International Conference on Machine Learning, 2017 :PMLR, pp. 3299-3308 .

[8] M. Mahmoud *et al.*, *"Tensordash: Exploiting sparsity to accelerate deep neural network training,"* in 2020 53rd Annual IEEE/ACM International Symposium on Microarchitecture (MICRO), 2020: IEEE, pp. 781-795 .

[9] L. Liu *et al.*, *"Dynamic sparse graph for efficient deep learning,"* arXiv preprint arXiv:1810.00859, 2018.

[10] D. Yang, A. Ghasemazar, X. Ren, M. Golub, G. Lemieux, and M. Lis, *"Procrustes: a dataflow and accelerator for sparse deep neural network training,"* in 2020 53rd Annual IEEE/ACM International Symposium on Microarchitecture (MICRO), 2020: IEEE, pp. 711-724 .

[11] J. S. Lew, Y. Liu, W. Gong, N. Goli, R. D. Evans, and T. M. Aamodt, *"Anticipating and eliminating redundant computations in accelerated sparse training* ",in Proceedings of the 49th Annual International Symposium on Computer Architecture, 2022, pp. 536-551 .

[12] J. Zhang, X. Chen, M. Song, and T. Li, *"Eager pruning: Algorithm and architecture support for fast training of deep neural networks,"* in 2019 ACM/IEEE 46th Annual International Symposium on Computer Architecture (ISCA), 2019: IEEE, pp. 292-303 .

[13] L. Ning, H. Guan, and X. Shen, *"Adaptive deep reuse: Accelerating cnn training on the fly,"* in 2019 IEEE 35th International Conference on Data Engineering (ICDE), 2019: IEEE, pp. 1538-1549 .

[14] V. Janfaza *et al.*, *"MERCURY: Accelerating DNN Training By Exploiting Input Similarity,"* in 2023 IEEE International Symposium on High-Performance Computer Architecture (HPCA), 2023: IEEE, pp. 638-650 .

[15] V .Janfaza, K. Weston, M. Razavi, S. Mandal, and A. Muzahid, *"SIMCNN: Exploiting Computational Similarity to Accelerate CNN Training in Hardware,"* arXiv preprint arXiv:2110.14904, 2021.

[16] K. Hegde, J. Yu, R. Agrawal, M. Yan, M. Pellauer, and C. Fletcher" ,*UCNN: Exploiting computational reuse in deep neural networks via weight repetition,"* in 2018 ACM/IEEE 45th Annual International Symposium on Computer Architecture (ISCA), 2018: IEEE, pp. 674-687 .

[17] H. Falahati *et al.*, *"Data-Aware compression of neural networks,"* IEEE Computer Architecture Letters, vol. 20, no. 2, pp. 94-97, 2021.

[18] Y. Zhao *et al.*, *"Cambricon-Q: A hybrid architecture for efficient training,"* in 2021 ACM/IEEE 48th Annual International Symposium on Computer Architecture (ISCA), 2021 :IEEE, pp. 706-719 .

[19] J. Chen and X. Ran, *"Deep learning with edge computing: A review,"* Proceedings of the IEEE, vol. 107, no. 8, pp. 1655-1674, 2019.

[20] M. Rüb and A. Sikora, *"A Practical View on Training Neural Networks in the Edge,"* IFAC-PapersOnLine, vol. 55, no. 4, pp. 272-279, 2022.

[21] A. Krizhevsky, V. Nair, and G. Hinton, *"The CIFAR-10 dataset,"* online: http://www. cs. toronto. edu/kriz/cifar. html, vol. 55, no. 5, 2014.

[22] Y.-H. Chen, J. Emer, and V. Sze, *"Eyeriss: A spatial architecture for energy-efficient dataflow for convolutional neural networks,"* ACM SIGARCH computer architecture news, vol. 44, no. 3, pp. 367-379, 2016.

پنجمین کنفرانس ملی انفورماتیک ایران
پژوهشگاه دانشهای بنیادی، پردیس فرمانیه، تهران
۱۳ و ۱۴ دی ماه ۱۴۰۲

پیش‌بینی ابتلا به بیماریهای مزمن به کمک ماشین بردار پشتیبان دوقلو با قیود نرم

حمیده فدیشه‌ای [۱]، جلال‌الدّین نصیری [۲]، سهراب عفّتی [۳]

[۱] گروه ریاضی کاربردی، دانشکده علوم ریاضی، دانشگاه فردوسی مشهد، مشهد
ha.fadishehei@mail.um.ac.ir
[۲] استادیار، گروه ریاضی کاربردی، دانشکده علوم ریاضی، دانشگاه فردوسی مشهد، مشهد
Jnasiri@um.ac.ir
[۳] استاد، گروه ریاضی کاربردی، دانشکده علوم ریاضی، دانشگاه فردوسی مشهد، مشهد
s-effati@um.ac.ir

چکیده

بیماری‌های مزمن از چالش‌های جدّی مربوط به سلامت انسان می‌باشند. تشخیص بهنگام آن‌ها می‌تواند برای داشتن سبک زندگی سالم مفید باشد. روش‌های یادگیری ماشین از جمله ماشین بردار پشتیبان برای پیش‌بینی این بیماری‌ها قابل استفاده هستند. در این بررسی به کمک یک روش ماشین بردار پشتیبان دوقلو با قیود نرم، سعی شده‌است تا به کمک اطّلاعات پزشکی بیماران، پیش‌بینی شود که آیا آن‌ها به بیماری‌های مزمن، مبتلاخواهند شد یا خیر.

ماشین بردار پشتیبان عادی به دنبال یافتن دو ابرصفحهٔ موازی با بیشترین فاصله است به‌طوری‌که داده‌های دو طبقه در دوطرف آن‌دو قرار گیرند. در ماشین بردار پشتیبان دوقلو، دو ابرصفحه لزوماً موازی نیستند. هریک از دو ابرصفحه نزدیکترین فاصله را به داده‌های کلاس خود داشته و از کلاس مقابل دارای یک فاصلهٔ حداقلی می‌باشند. سرعت این روش بهتر است ولی نسبت به داده‌های نویزی عملکرد خوبی ندارد. در روش پیشنهادی با استفاده از بهینه‌سازی فازی، محدودیت‌های مسأله به‌صورت روابط فازی در نظرگرفته می‌شوند؛ با این‌کار فضای شدنی مسأله بهینه‌سازی درجه دوم گسترش یافته، به نمونه‌ها اجازهٔ تخطی از ابر صفحات داده‌شده و تأثیر داده‌های دورافتاده در تشخیص نهایی کاهش یافته‌است. الگوریتم پیشنهادی، برروی داده‌های بالینی پزشکی، اجرا ونتایج با روش‌های مشابه مقایسه شده‌اند روش پیشنهادی عملکرد بهتری داشته است.

کلمات کلیدی

یادگیری ماشین، ماشین بردار پشتیبان دوقلو، بهینه‌سازی محدّب، محدودیت‌های فازی.

۱- مقدمه

بیماری‌های مزمن به دلیل طولانی بودن دوران ابتلا، عوارض آزاردهنده‌ی مختلف و زمینه‌سازی برای مبتلا شدن به بیماری‌های عفونی دیگر از جمله کرونا، از مهمترین مسأله‌های بهداشتی در جهان هستند. در سال ۲۰۱۹، حدود ۶۳٪ از مرگ و میر جهانی ناشی از بیماریهای مزمن بوده است. رایج‌ترین این بیماری‌ها، فشار خون، قند خون، و بیماریهای قلبی هستند که از نتایج سبک زندگی نادرست می‌باشند [9]. عوارض ناشی از آنها باعث آسیب رسیدن به قلب، کلیه، مغز و چشم‌ها می‌شود که آثاری مخرّب روی کار و زندگی افراد خواهد داشت[3]. افزون بر موارد ذکر شده این بیماران در برابر بیماری‌های عفونی (مثلاً ویروس کرونا) ایمنی کمتری دارند[11]. مطابق گزارش‌های سال ۲۰۱۹، ۴۸٪ افراد مبتلا به این ویروس بیماری مزمن نیز داشته و احتمال بروز علایم شدید در آنها بیشتر است [4]. به همهٔ موارد فوق هزینه‌های گرانی که به دولت‌ها و خانواده‌ها تحمیل می‌شوند نیز، قابل تأمّل است. بنابراین تشخیص زودهنگام بیماری کمک قابل توجّهی به سلامت فرد و جامعه خواهد داشت.

پس از الگوریتمهای شبکهٔ عصبی، در سال ۱۹۹۵ ایدهٔ جدیدتری به عنوان ماشین بردار پشتیبان توسط وَپنیک و همکاران مطرح شد که در آن از روشهای بهینه سازی مسایل درجه دوم، برای جداسازی داده‌های دو طبقه از یکدیگر استفاده می‌کرد. مبنای کار یافتن دو ابَرصفحهٔ موازی با بیشترین فاصلهٔ ممکن بود که داده‌های دو طبقه در دوطرف این ابرصفحات قرار گیرند [10]. در طی سال‌ها انواع جدیدتر، دقیق‌تر و کاراتری از ماشین بردار پشتیبان ارائه شد. برخی از این روش‌ها در زیر بیان شده‌اند.

۲- پژوهش‌های پیشین

۲-۱- ماشین بردار پشتیبان

فرض کنید که $X = \{(x_i, y_i)\}_{i=1}^{m}$ مجموعهٔ نمونه‌های آموزشی با اندازهٔ m باشد که هر $x_i \in R^n$ یک نمونهٔ یادگیری دارای n ویژگی در فضای ورودی[1] است ، $y_i \in \{1, -1\}$ برچسب کلاس دادهٔ x_i و $A_{m_1 \times n}$ مجموعهٔ نمونه‌های با برچسب ۱-، $B_{m_2 \times n}$ مجموعهٔ نمونه‌های با برچسب ۱+ و $m_1 + m_2 = m$ باشند، اگر داده‌ها در فضای ورودی، به‌صورت خطی قابل جداسازی باشند، می‌توان ابرصفحه‌ای جداساز به فرم $w^T x + b = 0$ یافت که در آن w یک بردار n بعدی و b کمیتی عددی است. ماشین بردار

پشتیبان به جای یافتن یک ابرصفحهٔ جداکننده، به دنبال یافتن دوابرصفحهٔ موازی است که تا حدامکان از یکدیگر دور بوده و داده‌های دو طبقه در دوطرف آنها واقع باشند. با مدل‌سازی مسألهٔ بالا و حل مسألهٔ بهینه‌سازی (۱)، میتوان ابرصفحهٔ جداساز را مشخص کرد.

$$\min_{(\omega\,,b)} \frac{1}{2}\|\omega\|^2 \quad \text{(۱)}$$
$$S.t.\quad y_i(\omega^T x_i + b) \geq 1, i = 1,2,\dots,m.$$

نمایش ماتریسی مسألهٔ (۱) به‌فرم (۲) خواهد بود:

$$\min_{(\omega\,,b)} \frac{1}{2}\|\omega\|^2 \quad \text{(۲)}$$
$$S.t.\quad A\omega + e_1 b \geq e_1,$$
$$B\omega + e_2 b \leq -e_2$$

در (۲)، e_1 و e_2 بردارهای با درایه‌های واحد و متناسب با A و B هستند.

در حالتی‌که داده‌ها در فضای ورودی، به‌صورت خطی جداناپذیر هستند، مسألهٔ *svm* با حاشیهٔ نرم به‌صورت (۳) مطرح خواهد شد:

$$\min_{(\omega\,,b)} \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{m}\xi_i \quad \text{(۳)}$$
$$S.t.\quad y_i(\omega^T x_i + b) \geq 1 - \xi_i$$
$$\xi_i \geq 0,\ i = 1,2,\dots,m.$$

که در آن ξ_i متغیر لغزش برای کاهش تأثیر دادهٔ نویزی x_i است.

پس از نوشتن تابع لاگرانژ و در نظرگرفتن شرایط مکمل زاید و حل مسألهٔ (۳) ابرصفحهٔ جدااساز معلوم خواهد شد.

$$\min \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j x_i^T x_j - \sum_{k=1}^{m}\alpha_k \quad \text{(۴)}$$
$$S.t.\quad \sum_{i=1}^{m}\alpha_i y_i = 0\,,$$
$$0 \leq \alpha_i \leq C\ ,i = 1,2,\dots,m.$$

که α_i ضرایب لاگرانژ هستند و تابع تصمیم $D(x)$ طبق فرمول (۴) موقعیت دادهٔ جدید x را نسبت به ابرصفحه مشخص خواهد کرد:

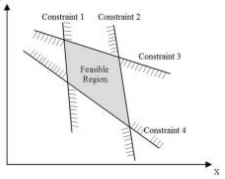
$$D(x) = sign(w^T x + b) \quad \text{(۵)}$$
$$= sign(\sum_{i\in S}\alpha_i y_i x_i^T x + b)$$

در رابطهٔ (۵)، S نشان‌دهندهٔ مجموعهٔ اندیس‌های بردارهای پشتیبان است. اگر نمونه‌ها در فضای ورودی جداپذیر خطی نباشند، نمونه‌ها با تکنیک حقهٔ هسته[۲] در فضای ویژگی با ابعاد بیشتر نگاشته می‌شوند. در این فضا، ماهیت خود داده ، اهمیتی ندارد بلکه آنچه مهم است فاصلهٔ داده ها از یکدیگر است. با این روش ماشین بردار پشتیبان یک ابرصفحهٔ جداکنندهٔ بهینه برای جداسازی نمونه‌ها پیدا می‌کند.

۲-۲- ماشین بردار پشتیبان با قیود نرم

مسألهٔ بهینه‌سازی (۶)، برنامه‌ریزی فازی نامتقارن نامیده می‌شود، اگر محدودیت‌هایش به‌صورت فازی بیان شوند. برای حل این مسأله روش‌هایی پیشنهاد شده است [۲] .

$$Optimize\ \ f(x)$$
$$S.t.\ \begin{cases} constraint\ 1 \\ constraint\ 2 \\ constraint\ 3 \\ constraint\ 4 \end{cases} \quad \text{(۶)}$$



شکل ۱: فضای شدنی برای مسألهٔ (۶)

در دسته‌بندی با قیدهای نرم، محدودیت‌های مسأله به فرم محدودیت فازی درمی‌آیند:

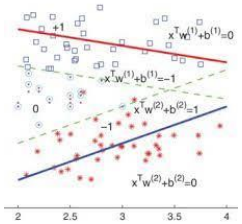
$$y_i(w^T x_i + b) \gtrsim 1 - \xi_i\,, \qquad i = 1,\dots,m. \quad \text{(۷)}$$

ماشین بردار پشتیبان با قیود نرم، انعطاف بیشتری نسبت به ماشین بردار پشتیبان عادی دارد؛ داده‌ها اجازهٔ تخطی از قیدها را دارند، با این راهکار، فضای شدنی مسألهٔ (۶) گسترش می‌یابد که در نتیجه تأثیر داده‌های نویزی کمتر می‌شود. توضیحات بیشتر در مرجع [۱] آمده‌اند.

۲-۳- ماشین بردار پشتیبان دوقلو

با گسترش ایدهٔ SVM در سال ۲۰۰۷ ، توسط جایادوا و همکاران ایدهٔ یافتن دوابرصفحهٔ غیرموازی، مطرح شد که هریک از آن‌ها تاحد ممکن به داده‌های یک طبقه نزدیک باشند و از طبقهٔ مقابل فاصلهٔ مطلوبی بگیرند. این الگوریتم‌ها به ماشین بردار پشتیبان دوقلو[۳] مشهورند. در این مسایل به جای حل یک مسألهٔ بهینه سازی درجه دوم با ابعاد بالا، دو مسألهٔ درجه دوم جدید با ابعاد تقریباً نصف ابعاد قبل(به شرط وجود تعادل بین داده‌های دو طبقه) و با مرتبهٔ محاسباتی $\frac{1}{4}$ روش قبل حل می‌شوند. این بالاترین مزیّت روش جدید به شمار می‌آید. گسترش‌های قابل توجّهی از ماشین بردار پشتیبان دوقلو مطرح [7] و کاربردهای متعدّدی از این روش ارائه شده است که از جملهٔ آن‌ها می‌توان به تشخیص رفتار انسانی [6] اشاره کرد.

شکل ۲: ماشین بردار پشتیبان دوقلو



در توضیح الگوریتم ماشین بردار پشتیبان دوقلو به‌فرض مجموعه‌ای از نمونه ها و اطلاعاتی درمورد آنها موجود است که به هر نمونه یک برچسب اختصاص داده‌شده و به کمک برچسب‌ها (۱ یا ۱−) میتوان کل نمونه ها را به دوگروه دسته‌بندی کرد. بهتر است که توزیع داده‌ها متعادل باشد یعنی با نسبت تقریباً مساوی، با برچسب ۱ یا۱− دسته‌بندی شده باشند. در روش بردار پشتیبان دوقلوبه جای یافتن دو ابرصفحهٔ موازی هم، ابرصفحه‌های جداساز، ما متقاطع اند. (مثلاً h_1 و h_2)

$$h_1 : x^T\omega_1 + b_1 = 0 \quad \text{(۸)}$$
$$h_2 : x^T\omega_2 + b_2 = 0$$

که در رابطه های فوق $\omega_1 \in R^n$ و ω_2 نمایشگر مختصات ابر صفحه b_1 و b_2 بایاس می‌باشند. h_1 تا حدّ امکان به کلاس ۱+ نزدیک و از کلاس ۱− دور است و h_2 تا حدّ ممکن به کلاس ۱− نزدیک و از کلاس ۱+ دور می‌باشد. داده‌های با برچسب منفی بایدفاصله‌ای مطلوب و حداقلی (مثلا یک واحد) از h_1 داشته باشند در غیراین‌صورت خطای دسته‌بندی ایجاد خواهد شد و لازم است که پارامترهای مشخص‌کنندهٔ جداساز طوری تنظیم شوند که داده‌هایی از این دست کمترخ دهند. مشابها در مورد داده‌های دستهٔ دوم هم چنین وضعیتی رخ خواهد داد. از این رو دو مسألهٔ مینی‌موم سازی مطرح خواهد شد و هریک از این دو مسأله نیاز به کمینه کردن دو مقدار متفاوت دارند؛ کم کردن فاصلهٔ ابرصفحهٔ جداساز h_1 از کلاس ۱+ و کم کردن تعداد نمونه‌هایی از کلاس ۱− که به h_1 نزدیک شده اند. تمام مباحث

گفته‌شده در مورد جداساز h_2 هم صدق خواهند کرد. با توجه به اینکه مسألهٔ بهینه‌سازی درجه دوم (QPP) به فرم استاندارد زیر است:

$$minimize \quad \left(\frac{1}{2}\right) x^T P x + q^T x \qquad (۹)$$

$$S.t. \quad \begin{cases} Gx \le h \\ Ax = b \end{cases}$$

دو مسألهٔ بهینه‌سازی به صورت زیر تعریف می‌شوند:

$$\min_{(\omega_1 , b_1)} \quad \frac{1}{2} \|A\omega_1 + e_1 b_1\|^2 + C_1 e_2^T q_1 \qquad (۱۰)$$
$$S.t. \quad -(B\omega_1 + e_2 b_1) + q_1 \ge e_2 , q_1 \ge 0$$

و

$$\min_{(\omega_1 , b_1)} \quad \frac{1}{2} \|B\omega_2 + e_2 b_2\|^2 + C_2 e_1^T q_2 \qquad (۱۱)$$
$$S.t. \quad (A\omega_2 + e_1 b_2) + q_2 \ge e_1 , q_2 \ge 0$$

که در آن ، $q_1 \in \mathcal{R}^{m_1}$، $q_2 \in \mathcal{R}^{m_2}$ وC_2 و C_1 پارامترهای خطا با مقادیر مثبت و سایر متغیرها قبلا تعریف شده‌اند. برای حل مسأله‌های فوق، بعد از نوشتن دوگان لاگرانژ و شرایط *مکمل زاید*[۴]، به دو مسألهٔ جدید زیر می‌رسیم:

$$\min_{\alpha} \quad \frac{1}{2} \alpha^T G (H^T H + \epsilon I)^{-1} G^T \alpha - e_2^T \alpha \qquad (۱۲)$$
$$S.t. \quad 0 \le \alpha \le C_1 e_2$$

و

$$\min_{\beta} \quad \frac{1}{2} \beta^t H (G^T G + \epsilon I)^{-1} H^T \beta - e_1^T \beta \qquad (۱۳)$$
$$S.t. \quad 0 \le \beta \le C_2 e_1$$

پس از بهینه‌سازی و یافتن بردارهای α وβ ، ابر صفحه‌های دو کلاس از دستورات زیر به دست می‌آیند:

$$\begin{bmatrix} \omega_1 \\ b_1 \end{bmatrix} = -(H^T H + \epsilon I)^{-1} G^T \alpha \qquad (۱۴)$$

$$\begin{bmatrix} \omega_2 \\ b_2 \end{bmatrix} = (G^T G + \epsilon I)^{-1} H^T \beta \qquad (۱۵)$$

در رابطهٔ (۱۵) ماتریسهای H و G به‌صورت $H = [A\ e_1]$ و G =[B e_2]

تعریف شده‌اند. مقدار کوچک و مثبت ε به قطر اصلی دوماتریس افزوده شده تا ماتریسها وارون‌پذیر باشند. در نهایت با حل مسأله فوق و یافتن ابرصفحات جداکننده، فاصلهٔ نمونه‌های جدید از هر ابر صفحه محاسبه شده و مینی‌موم فاصله، مشخص خواهد کرد که نمونهٔ ذکر شده به کدام دسته تعلق دارد.

در صورتیکه داده‌ها بصورت خطی قابل تفکیک نباشند، شبیه SVM از حقه‌ی هسته استفاده شده و ابرصفحه هایی با معادله‌های زیر تشکیل خواهند شد:

$$h_1 : K(x^T, C^t)\omega_1 + b_1 = 0 \qquad (۱۶)$$

$$h_2 : K(x^T, C^t)\omega_2 + b_2 = 0$$

توضیحات بیشتر در مورد داده‌هایی که به‌طور خطی جداناپذیرند، در مرجع [8]ذکر شده‌است.

۳- راه حلّ پیشنهادی: ماشین بردار پشتیبان دوقلو با قیود نرم[۵]

۳-۱- طبقه‌بند خطی

در این بررسی تلاش خواهدشد تا به کمک یک نوع ماشین بردار پشتیبان دوقلو با محدودیت‌های نرم، ابتلای فرد به بعضی از انواع بیماری‌های مزمن، پیش‌بینی شود. روش پیشنهادی، در مقایسه با ماشین بردار پشتیبان دوقلو، هنگام مواجهه با داده‌های نویزی عملکرد بهتری دارد.

در روش پیشنهادی سعی شده‌است تا با ایجاد تغییر در محدودیت‌های مسألهٔ بهینه سازی درجه دوم در ماشین بردار پشتیبان دوقلو، به‌جای نامساوی‌های ساده از نامساوی‌های فازی استفاده شود. مسأله‌های بهینه‌سازی جدید از قرار زیر هستند:

$$\min_{(\omega_1 , b_1)} \quad \frac{1}{2} \|A\omega_1 + e_1 b_1\|^2 + C_1 e_2^T q_1 \qquad (۱۷)$$
$$S.t. \quad -(B\omega_1 + e_2 b_1) + q_1 \gtrsim e_2 , q_1 \ge 0$$

و

$$\min_{(\omega_2 , b_2)} \quad \frac{1}{2} \|B\omega_2 + e_2 b_2\|^2 + C_2 e_1^T q_2 \qquad (۱۸)$$
$$S.t. \quad (A\omega_2 + e_1 b_2) + q_2 \gtrsim e_1 , q_2 \ge 0$$

علامت $\gtrsim$ به این معنی است که برای نمونه‌های یادگیری اجازه‌ی تخطی از حدود هم داده می‌شود. در واقع با این‌کار فضای شدنی مسأله گسترش می‌یابد و جواب‌های بهینه از فضای بزرگتری می‌توانند انتخاب شوند.

ابتدا می‌بایست دو مسألهٔ بهینه‌سازی (۱۸) و (۱۷) که مسأله‌های بهینه‌سازی غیرخطی درجهٔ دوم با تابع هدف محدب و قیود نامساوی فازی هستند، حل شوند. در گام اوّل برای حل مسأله، محدودیتها را از فرم ماتریسی خارج کرده و برای هر نمونه از هر طبقه ۱-، یک محدودیت به شکل (۱۹) در نظر گرفته می‌شود.

$$-(\omega_1{}^T x_i + b_1) \gtrsim 1 - q_{1_i} , \ q_{1_i} \ge 0 \qquad (۱۹)$$
$$i = 1, \dots, m_2.$$

که x_i، i_ اُمین نمونه از داده‌های با برچسب ۱- و q_{1_i} خطای مربوط به این نمونه در اثر تخطی از محدودیت نظیر و نزدیک‌شدن به h_1 می‌باشد. به‌طور مشابه برای نمونهٔ i_ام از طبقهٔ ۱+ محدودیت به شکل (۲۰) خواهد بود:

$$(\omega_2{}^T x_i + b_2) \gtrsim 1 - q_{2_i}, \ q_{2_i} \ge 0$$
$$i = 1, \dots, m_1. \qquad (۲۰)$$

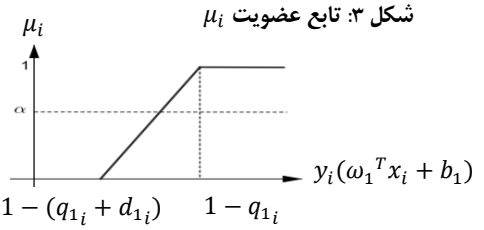
که x_i، i_ اُمین نمونه از داده‌های با برچسب ۱+ و q_{2_i} خطای مربوط به این نمونه در اثر تخطی از محدودیت نظیر و نزدیک‌شدن به h_2 می‌باشد. برای نمونه‌های با برچسب ۱+ طرفین محدودیت نظیر در (۱+) و برای نمونه‌های با برچسب ۱-، طرفین محدودیت نظیر در (۱-) ضرب شده و در نهایت محدودیت‌ها به شکل (۲۱) ادغام خواهندشد.

$$y_i(\omega_1{}^T x_i + b_1) \gtrsim 1 - q_{1_i}, \qquad (۲۱)$$
$$q_{1_i} \ge 0 , \ i = 1, \dots, m_2.$$

و $y_i(\omega_2{}' x_i + b_2) \gtrsim 1 - q_{2_i}$, $q_{2_i} \geq 0\,, i = 1, \dots, m_1.$

گام دوم تعیین توابع عضویت خطی $\mu_i\,, \mu'_i$ برای نامساوی‌های فازی موجود در قیود مسأله است. با توجه به شکل (۲)، در مورد اولین محدودیت در رابطهٔ (۲۱) و با تمرکز بر داده‌های کلاس ۱−، تابع عضویت μ_i مربوط به این طبقه تفصیل بررسی خواهد شد. μ'_i مشابه خواهد بود.

شکل ۳: تابع عضویت μ_i



طبق رابطهٔ (۲۱) هریک از نمونه‌های فوق، یک محدودیت به مسأله تحمیل می‌کنند. مجموعهٔ این محدودیت‌ها فضای شدنی را می‌سازد. متغیر−های مسأله یعنی $(\omega_1, b_1, q_1) \in R^{m_2+1+n}$ از این فضا انتخاب می‌شوند. هنگام حل مسأله این شرایط باید برآورده شوند. استفاده از نامساوی فازی به این محدودیت‌ها انعطاف بیشتری می‌بخشد. پارامترهای α و d_{1_i} توسط کاربر تعیین می‌شوند، پارامتر α ، سطحی را مشخص می‌کند که برای سطوح پایین‌تر از آن میزان تعلق فازی برای نامساوی موجود در قیود (۲۱)، صفر است. هر چه این مقدار به عدد ۱ نزدیک‌تر باشد، قیود مسأله به TWSVM نزدیک‌تر می‌شود. d_{1_i} ها (که در اینجا مقدارشان مساوی d_1 است)، مقدار تحملی هستند که می‌توان به نمونه‌ها نسبت داد. هرچه مقدار تحملی که نسبت می‌دهیم بیشتر باشد، نمونه‌های کلاس ۱− می‌توانند به ابرصفحهٔ جداساز کلاس ۱+ نزدیک‌تر شوند. این بدان معنی است که تعداد بردارهای پشتیبان بیشتر شده و داده‌های بیشتری در تعیین ابرصفحهٔ بهینه نقش خواهند داشت. تعریف می‌کنیم:

$$\mu_i: R^{m_2+1+n} \to (0,1],\quad i = 1,2,\dots,m_2. \qquad (۲۲)$$

$$\mu_i(\omega_1, b_1, q_1) = \begin{cases} 1, & if \quad y_i(\omega_1{}^T x_i + b_1) \geq 1 - q_{1_i} \\ \frac{y_i(\omega_1{}^T x_i + b_1) - 1 + q_{1_i} + d_{1_i}}{d_{1_i}} & if \quad 1 - (q_{1_i} + d_{1_i}) \leq y_i(\omega_1{}^T x_i + b_1) \leq 1 - q_{1_i} \\ 0, & if \quad y_i(\omega_1{}^T x_i + b_1) \leq 1 - (q_{1_i} + d_{1_i}) \end{cases}$$

و برای هر قید (۲۱) تعریف می‌کنیم:

$$X_i = \{(\omega_1, b_1, q_1) \in R^{m_2+1+n} \mid y_i(\omega_1{}^T x_i + b_1) \gtrsim 1 - q_{1_i}\,, q_{1_i} \geq 0\,,\quad i = 1, \dots, m_2\} \qquad (۲۳)$$

آن‌گاه جواب‌های مسأله باید از مجموعهٔ X با تعریف زیر انتخاب شوند:

$$X = \cap_{i \in I} X_i\,,\quad I = \{1,2,\dots,m_2\} \qquad (۲۴)$$

بنابراین مسألهٔ *rtwsvm1* را می‌توان به‌صورت زیر نوشت:

$$Minimize\ \{Q(\omega_1, b_1, q_1) = \frac{1}{2}\|A\omega_1 + e_1 b_1\|^2 + C_1 e_2^T q_1 \mid (\omega_1, b_1, q_1) \in X\} \qquad (۲۵)$$

برای حل (۲۵) از روش برش آلفا[۶] استفاده می‌شود. این مسأله در شکل ۳ دیده می‌شود. جواب‌های (۲۵) باید از مجموعهٔ

$$X_\alpha = \{(\omega_1, b_1, q_1) \in R^{m+1+n} \mid \mu_X(\omega_1, b_1, q_1) \geq \alpha\} \qquad (۲۶)$$

انتخاب شوند که

$$\mu_X(x) = \inf\{\mu_i(x), i \in I\} \qquad (۲۷)$$

مجموعهٔ X_α آلفا برش قید i ام است و $\alpha \in (0\,, 1]$.

جواب بهینهٔ مسألهٔ (۱۷) با α داده‌شده برابر است با:

$$S(\alpha) = \{(\omega_1, b_1, q_1) \in R^{m_2+1+n} \mid \frac{1}{2}\|A\omega_1 + e_1 b_1\|^2 + C_1 e_2^T q_1\} = \{Min \frac{1}{2}\|A\omega'_1 + e_1 b'_1\|^2 + C_1 e_2^T q'_1\,, (\omega'_1, b'_1, q'_1) \in X_\alpha\} \qquad (۲۸)$$

پس می‌توان نوشت:

$$X_\alpha = \cap_{i \in I}\{(\omega_1, b_1, q_1) \in R^{m_2+1+n} \mid y_i(\omega_1{}^T x_i + b_1) \geq r_{1_i}(\alpha), q_{1_i} \geq 0\,, i = 1, \dots, m_2\} \qquad (۲۹)$$

که $r_{1_i}(\alpha) = 1 - q_{1_i} - d_{1_i}(1-\alpha)$.

زیرا:

$$\frac{y_i(\omega_1{}^T x_i + b_1) - 1 + q_{1_i} + d_{1_i}}{d_{1_i}} \geq \alpha \Rightarrow y_i(\omega_1{}^T x_i + b_1) \geq 1 - q_{1_i} - d_{1_i}(1-\alpha) \qquad (۳۰)$$

در نهایت مدل RTWSVM عبارت‌است از:

$$Minimize \quad \frac{1}{2}\|A\omega_1 + e_1 b_1\|^2 + C_1 e_2^T q_1 \qquad (۳۱)$$
$$S.t. -(B\omega_1 + e_2 b_1) \geq e_2 - q_1 - d_1(1-\alpha)$$
$$q_{1_i} \geq 0\,,\quad i = 1, \dots, m_2.$$

گام بعدی حل مسألهٔ بهینه‌سازی غیرخطی درجه دوم با قیدهای غیرفازی۷ است. تابع هدف اولّیه همراه با محدودیت‌هایش، به تابع لاگرانژ تبدیل می‌شوند:

$$Q(\omega_1, b_1, q_1, \beta_1, \gamma_1) = \frac{1}{2}\|A\omega_1 + e_1 b_1\|^2 + C_1 e_2^T q_1 - \beta_1{}^T\{-(B\omega_1 + e_2 b_1) + q_1 + d_1(1-\alpha) - e_2\} - \gamma_1{}^T q_1 \qquad (۳۲)$$

که در آن $\gamma_1 = (\gamma_{1_1}, \dots, \gamma_{1_{m_2}})^T$ ، $\beta_1 = (\beta_{1_1}, \dots, \beta_{1_{m_2}})^T$ ضرایب لاگرانژ و نامنفی می‌باشند.

$$\frac{\partial Q(\omega_1, b_1, q_1, \beta_1, \gamma_1)}{\partial \omega_1} = 0 \Rightarrow A^T(A\omega_1 + e_1 b_1) + B^T \beta_1 = 0 \qquad (۳۳)$$

$$\frac{\partial Q(\omega_1, b_1, q_1, \beta_1, \gamma_1)}{\partial b_1} = 0 \Rightarrow e_1{}^T(A\omega_1 + e_1 b_1) + e_2{}^T \beta_1 = 0 \qquad (۳۴)$$

$$\frac{\partial Q(\omega_1, b_1, q_1, \beta_1, \gamma_1)}{\partial q_1} = 0 \Rightarrow C_1 e_2^T - \beta_1{}^T -, \gamma_1{}^T = 0 \qquad (۳۵)$$

به‌علاوه باید شرایط مکمل زاید نیز برقرار شوند:

$$\beta_1{}^T\{-(B\omega_1 + e_2 b_1) + q_1 + d_1(1-\alpha) - e_2\} = 0 \quad , \gamma_1{}^T q_1 = 0 \qquad (۳۶)$$

از ترکیب روابط (۳۳) و (۳۴) :

$$U_1 = -(H^t H + \epsilon I)^{-1} G^t \beta_1 \quad (۳۷)$$

که در آن: $G = [B\ e_2], H = [A\ e_1]$ و $U_1 = [\omega_1\ b_1]^T$. با جایگذاری (۳۷) در (۳۲) و ساده کردن روابط و استفاده از رابطهٔ (۳۵)، مسألهٔ نهایی به فرم زیر خواهد بود:

$$Min \frac{1}{2} \beta_1{}^T G(H^T H + \epsilon I)^{-1} G^T \beta_1 - \beta_1{}^T \{d_1(1-\alpha) - e_2\} \quad (۳۸)$$

$$S.t. \qquad 0 \le \beta_1 \le C_1 e_2$$

مشابه تمام روابط (۲۲) تا (۳۶) برای داده‌های کلاس دیگر نیز برقرار است. با حل (۳۸) و به‌دست‌آمدن ضرایب لاگرانژ و مشخص شدن دو ابرصفحهٔ غیرموازی، تابع تصمیم (۳۹) مشخص خواهد کرد که نمونهٔ جدید به کدام طبقه تعلق می‌گیرد.

$$\underset{j=1,2}{argmin} \ |x^T \omega_j + b_j| \quad (۳۹)$$

۳-۲- طبقه‌بند غیرخطی

توسعه‌ی روش پیشنهادی RTWSVM برای حالت غیرخطی مشابه *SVM* استاندارد است. برای این حالت، داده‌های آموزش با استفاده از تابع φبه فضای ویژگی نگاشت می‌شوند. با تعریف تابع هستهٔ:

$$K(x_i, x_j) = \varphi^T(x_i)\varphi(x_j) = z_i . z_j$$

ابرسطح‌های جداکننده به فرم (۱۶) تبدیل می‌شوند. در فضای ویژگی مسألهٔ اولیه به صورت (۴۰) و (۴۱)

$$\underset{(\omega_1, b_1)}{min} \ \frac{1}{2} \|K(A, C^T)\omega_1 + e_1 b_1\|^2 + C_1 e_2^T q_1 \quad (۴۰)$$
$$S.t. \quad -(k(B, C^T)\omega_1 + e_2 b_1) + q_1 \gtrsim e_2, \quad q_1 \ge 0$$

$$\underset{(\omega_2, b_2)}{min} \ \frac{1}{2} \|K(B, C^T)\omega_2 + e_2 b_2\|^2 + C_2 e_1^T q_2 \quad (۴۱)$$
$$S.t. \quad (K(A, C^T)\omega_2 + e_1 b_2) + q_2 \gtrsim e_1, \quad q_2 \ge 0$$

که در آن $C = [A\ B]$ و K تابع هستهٔ دلخواه می‌باشد.

مسأله برای (۴۰) ادامه خواهد یافت. برای (۴۱) روش‌ها مشابه خواهد بود.

$$\underset{}{min} \ \frac{1}{2} \|K(A, C^T)\omega_1 + e_1 b_1\|^2 + C_1 e_2^T q_1 \quad (۴۲)$$
$$S.t. \quad -(K(B, C^T)\omega_1 + e_2 b_1) \ge e_2 - q_1 - d_1(1-\alpha)$$
$$q_{1_i} \ge 0, \quad i = 1, \dots, m_2.$$

و مسأله‌ی دوگان بصورت (۴۳) درمی‌آید.

$$Q(\omega_1, b_1, q_1, \beta_1, \gamma_1) = \frac{1}{2} \|K(A, C^T)\omega_1 + e_1 b_1\|^2 + C_1 e_2^T q_1 - \beta_1{}^T \{-(K(B, C^T)\omega_1 + e_2 b_1) + q_1 + d_1(1-\alpha) - e_2\} - \gamma_1{}^T q_1 \quad (۴۳)$$

پس از مشتق‌گیری و برقراری شرایط مکمل زاید:

$$U_1 = -(S^T S + \epsilon I)^{-1} R^T \beta_1 \quad (۴۴)$$

که در آن: $R = [K(B, C^T)\ e_2], S = [K(A, C^T)\ e_1]$

$U_1 = [\omega_1\ b_1]^T$ و

مشابه نسخهٔ خطی کلاس نمونهٔ جدید x با محاسبهٔ فاصلهٔ عمودی آن از دو ابرسطح مشخص می‌شود ؛ به طوری که تابع تصمیم برای نسخهٔ غیرخطی به‌صورت زیر تعریف می‌گردد:

$$\underset{j=1,2}{argmin} \ |K(x^T, C^T)\omega_j + b_j| \quad (۴۵)$$

۴- نتایج و ارزیابی

۴-۱- پایگاه داده

داده‌های مورد استفاده در این تحقیق، از پایگاه‌های دادهٔ Kaggle و UCI استخراج شده‌اند. به دلیل اهمیت موضوع چهار مجموعه دادهٔ :CKD، Hep، PIDD و WDBC که به ترتیب مرتبط با بیماری کلیه، هپاتیت، دیابت و سرطان سینه هستند، مورد بررسی قرارگرفته‌اند. داده‌های توصیفی از دست رفته با اولین مُد در ستون مزبور و داده‌های عددی با صفر جایگزین شده‌اند. جدول (۱) مجموعهٔ داده را توصیف می‌کند.

مجموعهٔ داده	تعداد نمونه‌ها	تعداد نمونه‌های مثبت	تعداد نمونه‌های منفی	تعداد ویژگی	دادهٔ از دست‌رفته
CKD	۴۰۰	۲۴۸	۱۵۲	۲۴	ندارد
Hep	۱۵۵	۳۲	۱۲۳	۱۹	دارد
PIDD	۷۶۸	۲۶۸	۵۰۰	۸	ندارد
WDBC	۵۶۹	۲۱۲	۳۵۷	۳۰	دارد

جدول ۱: مشخصات مجموعهٔ داده برای ارزیابی روش RTWSVM

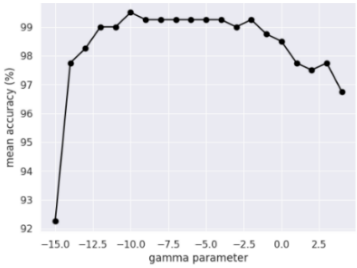
۴-۲- نحوهٔ پیاده‌سازی و اجرای الگوریتم‌ها

تمام روش‌ها در زبان برنامه‌نویسی پایتون در محیط آنلاین google colab پیاده‌سازی شده‌است. آزمایش‌ها روی یک کامپیوتر شخصی با پردازندهٔ core i5، سیستم عامل ویندوز ۱۰ و ۸ گیگابایت حافظه صورت گرفته است. از کتابخانهٔ cvxopt برای بهینه‌سازی استفاده شده است.

از آن‌جا که در دنیای واقعی داده‌ها جداپذیر خطی نیستند، از تابع هستهٔ گوسی که قدرت تعمیم‌پذیری بهتری دارد استفاده شده است:

$$K(x, x') = \exp\left(-\frac{\|x - x'\|^2}{2\gamma^2}\right) \quad (۴۶)$$

دقت روش RTWSVM به انتخاب پارامترهای بهینه بسیار وابسته است. بدین منظور از روش جستجوی شبکه‌ای برای پیداکردن پارامتر بهینه استفاده شده است. پارامترهای خطا از مجموعهٔ $\{2^i | i = -7, -6, \dots, 7\}$ وپارامترهای α_1 ، α_2 ، d_1 و d_2که به ترتیب، برای سطح برش آلفا و ضریب تحمل الگوریتم، مورد استفاده بوده‌اند از مجموعهٔ [۰.۹۵ و ۰.۷۵ و ۰.۲۵] و پارامتر تابع هسته γ نیز از $\{2^i | i = -15, -6, \dots, 5\}$ انتخاب شده‌اند. برای تعیین میزان دقّت روش، از اعتبارسنجی متقابل استفاده شده‌است؛ الگوریتم ارائه شده با ۹۰٪ داده‌ها (k=10) آموزش دیده و با ۱۰٪ باقیمانده آزمایش شده، این مراحل ده بار تکرار شده و پس از ۱۰ بار تکرار، میانگین و انحراف معیار دقت روش پیشنهادی به‌دست آمده‌اند. شکل ۵ بیان‌گر تأثیر تغییرات پارامتر گاما بر میزان دقت روش پیشنهادی در پایگاه دادهٔ CKD است.



شکل ۴: اثر افزایش پارامتر γ روی دقت دسته‌بند *rtwsvm*

٤-۳- نتایج تجربی

روش پیشنهادی با SVM ساده، رگرسیون خطی، درخت تصمیم و نزدیک‌ترین همسایه، که از مرجع [12] به‌دست‌آمده، مقایسه شده‌اند.

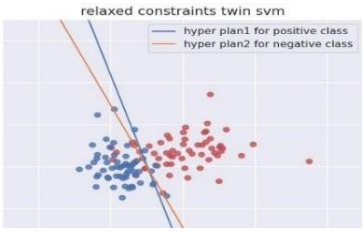
نتایج مقایسهٔ دقت و رتبهٔ الگوریتم پیشنهادی در جدول شماره ۲ و۳ قابل مشاهده اند؛ RTWSVM عملکرد بهتری دارد.

دقت	SVM	LR	DT	KNN	RTWSVM
CKD	۹۷.۸۵±۱.۵۵	۹۸.۷۵	۹۷.۵۰	۹۵.۰۰	۹۹.۵۰±۱
Hep	۸۰.۰۰	۸۳.۰۰	۸۳.۰۰	۸۰.۰۰	۱۰۰±۰
PIDD	۷۷.۹۲	۷۹.۸۷	۷۴.۶۸	۷۶.۶۲	۷۸.۸۱±۵.۵۹
WDBC	۹۷.۱۴	۹۷.۱۴	۹۵.۰۰	۹۷.۱۴	۹۸.۳۹±۱.۴۸
میانگین	۸۸.۲۲	۸۹.۶۹	۸۷.۵۴	۸۷.۱۹	**۹۴.۱۷**

جدول ۲: مقایسهٔ دقت دسته‌بندهای پیشین و روش پیشنهادی

رتبه	SVM	LR	DT	KNN	RTWSVM
CKD	۳	۲	۴	۵	۱
Hep	۴	۲	۲	۴	۱
PIDD	۳	۱	۵	۴	۲
WDBC	۳	۳	۵	۳	۱
میانگین	۳.۳۷۵	۲.۱۲۵	۴.۱۲۵	۴.۱۲۵	**۱.۲۵**

جدول ۳: مقایسهٔ رتبهٔ دسته‌بندهای پیشین و روش پیشنهادی



شکل ۵: نمایش هندسی بردار پشتیبان دوقلو با قیود نرم.

۵- نتیجه‌گیری و پژوهش‌های آینده

گسترش فضای شدنی، عدم تأثیر منفی بر تابع هدف، کاهش اثر داده‌های پرت، سرعت بیشتر و کارایی بالاتر از ویژگی‌های مثبت روش جدید است. بکارگیری الگوریتم‌های جستجو مانند الگوریتم‌های تکاملی، می‌تواند یافتن مقدار بهینه برای پارامترهای زیاد موجود را سرعت بخشد. نیز می‌توان با در نظرگرفتن یک ماتریس وزنی ضرایب تحمل، به هریک از نمونه‌ها توجه ویژه داشت. با این کار تأثیر داده‌های نویزی کاهش خواهد یافت.

مراجع

[۱] سبزه‌کار، مصطفی، **بررسی تأثیر توجه مضاعف به نمونه‌های یادگیری با استفاده از قیود بهینه‌سازی طبقه‌بندهای ماشین بردار پشتیبان،** پایان‌نامهٔ کارشناسی ارشد، گروه مهندسی کامپیوتر، دانشکده مهندسی، دانشگاه فردوسی مشهد، صفحات ۹۰-۸۲ ، آذر ۸۸.

[۲] ناجی عظیمی، زهرا، **آشنایی با برنامه‌ریزی خطی فازی،** ویراستهٔ وحیدیان کامیاد، علی، ویرایش اوّل، مشهد، انتشارات دانشگاه فردوسی مشهد، تابستان ۹۵.

[3] Alkenani, A. H., Li, Y., Xu, Y. & Zhang, Q. "Predicting Alzheimer's disease from spoken and written language using fusion-based stacked generalization", J. Biomed. Inform. 118, 103803, 2021.

[4] Guo, Y. et al. "A review of wearable and unobtrusive sensing technologies for chronic disease management", Comput. Biol. Med. 129, 104163, 2020.

[5] Higgins, V., Sohaei, D., Diamandis, E. P. & Prassas, I. "COVID-19: From an acute to chronic disease? Potential long-term health consequences", Crit. Rev. Clin. Lab. Sci. 58(5), 297–310, 2021.

[6] Mozafari, Kourosh, Jalal A. Nasiri, Nasrollah Moghadam Charkari, and Saeed Jalili. "Action recognition by local space-time features and least square twin SVM (LS-TSVM)." In *2011 first international conference on informatics and computational intelligence*, pp. 287-292. IEEE, 2011.

[7] Nasiri, Jalal A., and Amir M. Mir. "An enhanced KNN-based twin support vector machine with stable learning rules." *Neural computing and applications* 32, no. 16 : 12949-12969, 2020.

[8] Raschka S, Mirjalili V**.** Python machine learning second edition, Packt Publishing, BIRMINGHAM – MUMBA, 2017.

[9] Souza-Pereira, L., Pombo, N., Ouhbi, S., Felizardo, V. & Garcia, N. "Clinical decision support systems for chronic diseases: A systematic literature review", Comput. Methods Progr, Biomed, 195, 105565, 2020.

[10] Tanveer M, Rajani T, Rastogi R, Shao YH, Ganaie MA. "Comprehensive review on twin support vector machines", Annals of Operations Research, 8:1-46, 2022.

[11] Yuan, X., Chen, S., Yuwen, L., An, S., Mei, S. & Chen, T. "An improved SEIR model for reconstructing the dynamic transmission of COVID-19", In Proceedings of IEEE International Conference on Bioinformatics and Biomedicine (BIBM) 2320–2327 , 2020.

[12] Yuan X, Chen S, Sun C, Yuwen L. A novel early diagnostic framework for chronic diseases with class imbalance, Scientific Reports, 21;12(1):8614, 2022.

پانویس ها

۱ Input space

۲ Kernel Trick

۳ Twin Support Vector Machine (TWSVM)

۴ Karush–Kuhn–Tucker (KKT)

۵ Relaxed Constraints Twin Support Vector Machine (RTWSVM)

۶ $\alpha - cut$

۷ Crisp

پنجمین کنفرانس ملی انفورماتیک ایران
پژوهشگاه دانشهای بنیادی، پردیس فرمانیه، تهران
۱۳ و ۱۴ دی ماه ۱۴۰۲

پژوهش دکتری: پیش‌بینی تفسیرپذیر نتیجه‌ی فرایند کسب‌وکار

زهرا حسینی نژادمحبتی[1]، صادق علی‌اکبری[2]، رامک قوامی‌زاده میبدی[3]، حامد ملک[4]

[1] دانشکده مهندسی و علوم کامپیوتر، دانشگاه شهید بهشتی، تهران، ایران
z.hosseininezhad@sbu.ac.ir
[2] استادیار دانشکده مهندسی و علوم کامپیوتر، دانشگاه شهید بهشتی، تهران، ایران
s_aliakbary@sbu.ac.ir
[3] استادیار دانشکده مهندسی و علوم کامپیوتر، دانشگاه شهید بهشتی، تهران، ایران
R-Ghavami@sbu.ac.ir
[4] استادیار دانشکده مهندسی و علوم کامپیوتر، دانشگاه شهید بهشتی، تهران، ایران
h_malek@sbu.ac.ir

1. خلاصه‌ای از پژوهش

فرایند کسب وکار مجموعه‌ای از رویدادها، فعالیت‌ها و نقاط تصمیم‌گیری است که نهایتا منجر به یک نتیجه‌ی ارزشمند برای مشتری می‌شود [1]. سازمان‌ها با انبوهی از داده‌های حاصل از اجرای فرایندها مواجه هستند. اغلب در تحلیل این داده‌ها، به «فرایند»های درگیر در این داده‌ها توجه نمی‌شود. فرایندکاوی این شکاف میان مدل فرایند و داده‌های حاصل از اجرای فرایند را پر می‌کند و با ارائه روش‌های متعدد با اهدافی مانند کشف فرایند، بررسی انطباق میان فرایند مدل‌شده و اجرا شده، بهبود فرایند و پشتیبانی عملیات به دنبال افزودن دیدگاه فرایندی به تحلیل‌های داده‌کاوی است [2].

یکی از زیرشاخه‌های فرایندکاوی، پایش پیش‌بینانه‌ی فرایند کسب‌وکار است که به پیش‌بینی درباره‌ی وضعیت آینده‌ی فرایند می‌پردازد. وضعیت آینده‌ی فرایند شامل مواردی مانند زمان باقی‌مانده تا انتهای فرایند، فعالیت بعدی یا نتیجه‌ی نهایی اجرای فرایند است [3] که در این پژوهش، نتیجه مورد توجه قرار می‌گیرد.

برای هر فرایند و با توجه به اهداف کسب‌وکار می‌توان مجموعه‌ای از نتایج نهایی ممکن تعریف کرد. امروزه با پیشرفت‌های هوش مصنوعی و یادگیری ماشین، اغلب از این رویکردها در حل مسائل گوناگون استفاده می‌شود. پیش‌بینی نتیجه‌ی فرایند کسب‌وکار نیز یک مساله از نوع دسته‌بندی است که روش‌های متعدد یادگیری ماشین (مانند جنگل تصادفی و XGBoost) و یادگیری عمیق (مانند شبکه‌ی LSTM) برای حل آن ارائه شده‌اند. روش‌های یادگیری ماشین و مخصوصا یادگیری عمیق در کنار مزایایی مانند دقت بالا، ایراداتی نیز دارند. یکی از مهم‌ترین مشکلات این روش‌ها، ساختار جعبه‌سیاه آن‌ها است که باعث عدم شفافیت در رفتار مدل است. این عدم شفافیت در ماهیت رفتار مدل‌های جعبه‌سیاه سبب شده که استفاده از آن‌ها برای کاربردهای حساسی مانند پزشکی با مشکلاتی روبرو شود. هوش مصنوعی تفسیرپذیر ابزار موثری برای حل این مشکلات است و به مدل‌های یادگیری ماشین این توانایی را می‌دهد که رفتار یا تصمیمات خود را به نحوی که برای انسان قابل درک باشد، توضیح دهند [4].

بنابراین جنبه‌ی دیگری که در مساله‌ی پیش‌بینی نتیجه‌ی فرایند موردتوجه این پژوهش قرار می‌گیرد، تفسیر پیش‌بینی است. روشی که برای تفسیرپذیری استفاده می‌کنیم، ارتباط مستقیم با روشی دارد که برای پیش‌بینی نتیجه به کار برده‌ایم. در واقع ما ابتدا پیش‌بینی نتیجه را انجام می‌دهیم و بعد آن را تفسیر می‌کنیم. دو سوال اصلی برای ارائه‌ی تفسیر وجود دارد: چه چیزی به عنوان تفسیر در نظر گرفته می‌شود و چه رویکردهایی برای تفسیرپذیری براساس روش‌های پیش‌بینی نتیجه وجود دارد.

ارزیابی نیز چالش مهم دیگری است که در این پژوهش به آن می‌پردازیم، ارزیابی در این مساله دو جنبه دارد: ارزیابی کیفیت پیش‌بینی نتیجه، ارزیابی تفسیر ارائه شده برای این پیش‌بینی و بنابراین در ارزیابی نهایی هر دو جنبه مورد توجه ما قرار می‌گیرد.

کلمات کلیدی

فرایندکاوی، پایش‌پیش‌بینانه‌ی فرایند، پیش‌بینی نتیجه‌ی فرایند، پیش‌بینی تفسیرپذیر، یادگیری عمیق

2. شرح مسئله، پیشینه و اهمیت مسئله

فرایندکاوی یک حوزه‌ی تحقیقاتی میان رشته‌ای بین مدیریت فرایند کسب‌وکار و علم داده است که روی استخراج دانش از داده‌های حاصل از اجرای فرایندها تمرکز دارد. چرخه مدیریت فرایند کسب‌وکار شامل شناسایی فرایند، کشف فرایند، تحلیل فرایند، بازطراحی فرایند، پیاده‌سازی فرایند و پایش فرایند است. تکنیک‌های فرایندکاوی می‌توانند از فازهای مختلف این چرخه پشتیبانی کنند. پیش‌بینی نتیجه‌ی نهایی (زیرمجموعه‌ی پایش پیش‌بینانه‌ی فرایند کسب‌وکار) از مسائلی است که اخیرا مورد توجه زیادی قرار گرفته است. تعریف نتیجه‌ی فرایند کسب‌وکار توسط صاحب فرایند انجام شده و در راستای هدف فرایند است و می‌تواند دربرگیرنده‌ی طیف وسیعی از سوالات تحلیلی باشد، مثلا آیا نهایتا این فرایند با تاخیر مواجه می‌شود یا خیر؟ آیا با درخواست وام موافقت می‌شود یا خیر؟
گاهی آخرین فعالیت انجام شده در فرایند (مثلا فعالیت لغو یا تایید) به عنوان نتیجه قلمداد می‌شود و گاهی نیز نیاز به تعریف تابعی برای برچسب زدن نتیجه برحسب تحلیل موردنیاز وجود دارد.

نتیجه‌ی فرایند می‌تواند دودویی یا چندتایی باشد، البته نتیجه‌ی چندتایی نیز قابل تبدیل به چند نتیجه‌ی دودویی است، بنابراین در

پژوهش‌های پیشین، صرفا نتیجه‌های دودویی مدنظر قرار گرفته‌اند. در این پژوهش نیز نتیجه را به شکل دودویی در نظر می‌گیریم.

موضوع مهم دیگری که در این پژوهش مدنظر قرار می‌گیرد و در میان کارهای پیشین کمتر به آن پرداخته شده، توجه به تفسیر پیش‌بینی است. بنابراین یک جنبه‌ی مهم از پیش‌بینی نتیجه آن است که این پیش‌بینی در کنار داشتن ویژگی‌هایی مانند دقت بالا، تفسیرپذیر نیز باشد. به عنوان مثال در فرایند درخواست وام پیش‌بینی کند نتیجه منفی است و مشخص کند دلیل منفی بودن نتیجه، عدم وجود بعضی مستندات در یکی از گام‌های فرایند است.

به روش‌های پیشین برای حل مساله‌ی پیش‌بینی نتیجه از دو منظر می‌توان توجه کرد: اول اینکه از چه روشی برای پیش‌بینی نتیجه استفاده کرده‌اند، دوم اینکه آیا به تفسیر توجه کرده‌اند و اگر بله، چه نوع تفسیری ارائه کرده‌اند. همانطور که در جدول 1 مشاهده می‌شود، پژوهش‌های متنوعی به حل مساله‌ی پیش‌بینی نتیجه پرداخته‌اند که از نظر دقت با هم مقایسه شده‌اند، اما در تعداد کمی از این موارد به تفسیرپذیری توجه شده است. همچنین نوع تفسیری که تا الان ارائه شده، تعیین اهمیت ویژگی‌ها از طریق روش‌هایی مانند LIME یا وزن‌های مکانیزم توجه و تعیین فعالیت‌های مهم است. از طرفی در بعضی مقالات نشان داده شده مکانیزم توجه تفسیر خوبی ارائه نمی‌کند [17] و البته در در مقاله‌ی [12] که از مکانیزم توجه در LSTM استفاده کرده، تاکید روی دقت است و صرفا اشاره شده که وزن‌های مکانیزم توجه بیانگر اهمیت ویژگی‌ها هستند اما تفسیرپذیری هدف این مقاله نبوده است. در مورد مقاله‌ی [16] نیز یکی از اشکالات اصلی که در خود مقاله نیز اشاره شده، این است که برای تفسیر فقط به جریان کنترلی توجه کرده و مقادیر ویژگی‌ها را در نظر نگرفته است. بنابراین در مجموع با توجه به جدول 1 ارائه‌ی روشی که هم دقت بالا داشته باشد و هم تفسیر قابل قبولی ارائه دهد، یک چالش مهم است [6] و در این پژوهش مورد توجه ما قرار می گیرد.

جدول 1 روش‌های پیش‌بینی نتیجه‌ی فرایند با توجه به تفسیرپذیری

DT درخت تصمیم، LR رگرسیون منطقی، SVM ماشین بردار پشتیبان، RF جنگل تصادفی، LSTM حافظه طولانی کوتاه‌مدت، GRU واحد بازگشتی دروازه‌ای، CNN شبکه عصبی پیچشی، GGNN شبکه عصبی گراف دروازه‌ای

تقسیم‌بندی	روش	الگوریتم	پژوهش‌ها	دقت	تفسیرپذیری
جعبه سفید	یادگیری ماشین	DT	[7,8]	پایین	ذاتا تفسیرپذیر
		LR	[5]	پایین	
جعبه سیاه	یادگیری ماشین	SVM	[5]	متوسط	عدم توجه به تفسیر
		RF	[5, 8, 9, 11]	نسبتا بالا	در [11] تعیین اهمیت ویژگی‌ها
		XGBoost	[9-11]	نسبتا بالا	در [11] تعیین اهمیت ویژگی‌ها
	یادگیری عمیق	LSTM	[12]	بالا	وزن های مکانیزم توجه
		GRU	[13]	بالا	عدم توجه به تفسیر
		CNN	[14,15]	بالا	عدم توجه به تفسیر
		GGNN	[16]	بالا	تعیین فعالیت های مهم

اهمیت مساله‌ی پیش‌بینی نتیجه به این دلیل است که نتیجه‌ی فرایند، مرتبط با هدف فرایند و به طور کلی اهداف کسب‌وکار است، در صورتی که بتوان نتیجه را با دقت بالا و در زمان مناسبی پیش‌بینی کرد، زمان کافی برای اتخاذ اقدامات لازم جهت نیل به اهداف کسب‌وکار در اختیار داریم. همچنین وقتی تفسیری برای این پیش‌بینی ارائه می‌شود، اعتماد کاربران به چنین سامانه‌ای بیشتر می‌شود و در اتخاذ تصمیم کارامدتر نیز موثر است. به عنوان مثال در فرایند پذیرش بیمار در بیمارستان، اگر در زمان مناسب پیش‌بینی شود درصد بالایی از بیماران فعلی نیاز به مراقبت ویژه در آینده خواهند داشت، این پیش‌بینی باعث می‌شود منابع کافی مانند تعداد پرستار و اتاق مراقبت ویژه از قبل فراهم شوند. همچنین اگر تفسیری برای این پیش‌بینی نیز ارائه شود، کاربران می‌توانند بیش از پیش به این پیش‌بینی اعتماد کنند و تصمیمات جدی‌تری اتخاذ کنند.

3. نوآوری و برتری پروژه نسبت به پژوهش‌های مرتبط قبلی

در پژوهش‌های پیشین حوزه‌ی پایش‌بینانه، عمده‌ی پیش‌بینی‌ها در مورد زمان و فعالیت بعدی است و کارهای کمتری به پیش‌بینی نتیجه پرداخته‌اند. پیش‌بینی نتیجه به دلیل وابستگی به پارامترهای بیشتر، اساسا مساله دشوارتری است [5]. همچنین در بین تعداد کم پژوهش‌های انجام شده، عمدتا تفسیر موردتوجه قرار نگرفته یا مصادیق محدودی برای آن در نظر گرفته شده است. در این پژوهش به طور جدی به تفسیر نتیجه توجه می‌شود و به آن علاوه بر رویکردهای مرسوم در هوش مصنوعی تفسیرپذیر، از دریچه‌ی شباهت‌یابی نیز توجه شده که در بخش بعدی بیشتر تشریح می‌شود.

4. رویکرد و داده‌های استفاده‌شده

در مسائل فرایندکاوی، مهم‌ترین داده‌ی ورودی، نگاره رویداد است که حاوی تاریخچه‌ی اجرای فرایندها است. به هر نمونه‌ی اجرایی از فرایند، «نمونه» گفته می‌شود. هر نمونه از تعدادی «رویداد» تشکیل شده و هر رویداد بیانگر اجرای یک «فعالیت» مشخص توسط یک نمونه در یک «زمان» خاص است. بنابراین سه ویژگی اجباری هر رویداد، شناسه‌ی نمونه، فعالیت و برچسب زمانی است و هر سطر نگاره رویداد، متناظر با یک رویداد است. به جز این ویژگی‌ها ممکن است هر رویداد ویژگی‌های اختیاری دیگری مانند منبع (فرد یا دستگاه انجام‌دهنده‌ی فعالیت) نیز داشته باشد. نمونه‌ای از یک نگاره رویداد در جدول 2 نشان داده شده است که در آن سن و منبع دو ویژگی اختیاری هستند. ویژگی سن بین تمام رویدادهای نمونه مشترک بوده و منحصر به همان نمونه است، در حالی که منبع به ازای هر رویداد می‌تواند متفاوت باشد.

جدول 2 مثالی از یک نگاره رویداد

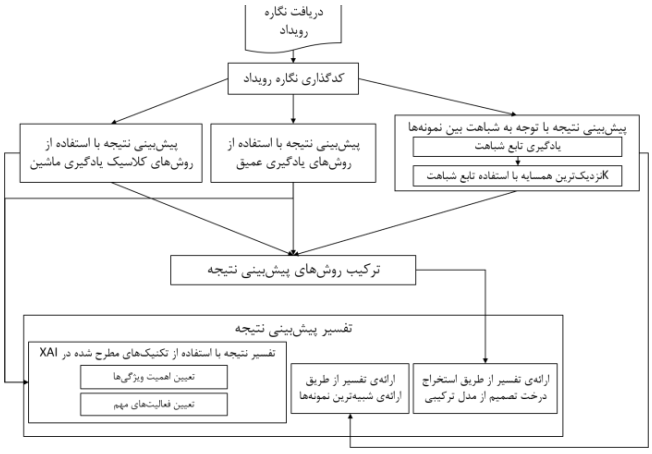
شناسه‌ی نمونه	فعالیت	برچسب زمانی	سن	منبع
1	ثبت درخواست	1402/8/1 9:50	25	Res_a
1	بررسی درخواست	1402/8/1 10:15	25	Res_b
1	تایید درخواست	1402/8/1 10:25	25	Res_c
2	ثبت درخواست	1402/8/3 8:20	22	Res_b
2	لغو درخواست	1402/8/3 8:20	22	Res_b

چارچوب کلی این پژوهش در شکل 1 ترسیم شده است. اولین گام، دریافت نگاره رویداد و سپس کدگذاری آن است. یکی از چالش‌های مهم در حل مسائل فرایندکاوی، کدگذاری داده‌ها است. همان‌طور که در جدول 2 مشاهده می‌شود، الزاما تعداد رویدادهای هر نمونه، تعداد یکسانی ندارند، اما هنگام آموزش مدل یادگیری ماشین یا یادگیری عمیق، بردارهای با طول یکسان نیاز داریم. برای حل این چالش راهکارهایی وجود دارد، به عنوان مثال می‌توان فقط تعداد مشخصی از رویدادها را در نظر گرفت و از اطلاعات سایر رویدادها صرف‌نظر کرد، راهکار دیگر در نظر گرفتن تمام رویدادها است، اما به جای استفاده از خود مقادیر ویژگی، از توابع تجمعی مانند میانگین، کمینه، بیشینه، تعداد رخداد و غیره برای هر ویژگی استفاده کنیم تا برای هر تعداد ویژگی، نهایتا یک مقدار وجود داشته باشد.

در قدم بعد، نتیجه را پیش‌بینی می‌کنیم که برای این کار سه رویکرد مختلف متصور هستیم، روش‌های کلاسیک یادگیری ماشین و روش‌های یادگیری عمیق که این دو رویکرد در کارهای پیشین نیز مورد توجه قرار گرفته‌اند، رویکرد سوم پیش‌بینی نتیجه با توجه به شباهت میان نمونه‌ها که در این پژوهش معرفی شده است. در پیش‌بینی نتیجه با توجه به شباهت نمونه‌ها، به جای آن که برای یک نمونه به طور مستقل به دنبال پیش‌بینی باشیم، به دنبال یافتن تابعی برای اندازه‌گیری شباهت میان نمونه‌ها هستیم، با در اختیار داشتن چنین تابعی، نمونه‌های مشابه نمونه‌ی جاری را داریم، بنابراین نتیجه برای نمونه‌ی جاری از طریق رای‌گیری میان نتایج نمونه‌های مشابه انجام می‌شود، با این فرض که چون این نمونه‌ها مشابه هستند، نتیجه‌ی نهایی نیز مشابه است. چنین روشی ذاتا تفسیرپذیر است، چون به عنوان تفسیر، به شباهت میان نمونه‌ها اتکا می‌کنیم. از نظر ادبیات هوش مصنوعی تفسیرپذیر نیز این نوع تفسیر معتبر و در واقع همان تفسیر مبتنی بر مثال است. در این نوع از تفسیر، مثال‌هایی از مجموعه داده انتخاب می‌شود تا رفتار مدل یادگیری ماشین با داده‌ی فعلی را توجیه کند [4]. شهود این نوع تفسیر این است که مثلا یک مدیر پروژه‌ی نرم‌افزار برای انتخاب تکنولوژی در پروژه‌ی نرم‌افزاری فعلی، از تجربیات خود در پروژه‌های مشابه قبلی استفاده می‌کند.

نوع تفسیر ارائه شده، ارتباط مستقیم با روشی دارد که برای پیش‌بینی نتیجه استفاده کرده‌ایم. بنابراین برای پیش‌بینی نتیجه با روش‌های کلاسیک یادگیری ماشین یا یادگیری عمیق از تفسیر با استفاده از روش‌های متداول در هوش مصنوعی تفسیرپذیر استفاده می‌کنیم. پیش‌بینی نتیجه با توجه به شباهت میان نمونه‌ها هم که ذاتا تفسیرپذیر است و به عنوان تفسیر، نمونه‌های مشابه ارائه می‌شوند.

با توجه به تنوع روش‌های موجود برای پیش‌بینی نتیجه، یکی دیگر از مشارکت‌های این پژوهش، تلاش برای ترکیب این روش‌ها است، البته چالشی که در اینجا وجود دارد این است که در صورت ترکیب روش‌ها برای پیش‌بینی نتیجه، تفسیر را نمی‌توان به راحتی ترکیب کرد. در این حالت پیشنهاد می‌شود یک درخت تصمیم از این مدل پیچیده‌ی حاصل از ترکیب سایر روش‌ها، استخراج شود. الگوریتم‌هایی برای این مورد در حوزه‌ی هوش مصنوعی تفسیرپذیر معرفی شده‌اند [18]. اگرچه تا حدی دقت درخت تصمیم استخراج شده نسبت به مدل اصلی کاهش پیدا می‌کند، اما در عوض درخت تصمیم ذاتا تفسیرپذیر است.



شکل 1 چارچوب کلی روش پیشنهادی

در این پژوهش دو نوع معیار ارزیابی مورد توجه قرار می‌گیرد: ارزیابی کیفیت پیش‌بینی نتیجه و ارزیابی کیفیت تفسیر. کیفیت پیش‌بینی از سه جنبه‌ی دقت، زودکرد و کارایی زمان قابل بررسی است. منظور از زودکرد این است که پیش‌بینی حتی‌الامکان در ابتدای کار فرایند انجام شود. برای ارزیابی دقت از معیار AUC (Area Under ROC Curve) استفاده می‌شود. ارزیابی تفسیرپذیری یک چالش مهم در هوش مصنوعی تفسیرپذیر [4] و فرایندکاوی [6] است. معیارهای اولیه مانند میزان سنجش وفاداری تفسیر به مدل وجود دارد، البته به دلیل چالش‌های فعلی ارزیابی تفسیرپذیری یک توصیه این است که برای ارزیابی تفسیر از دانش خبره [4] استفاده شود.

مجموعه داده‌های استفاده شده در این پژوهش از طریق 4TU.Centre for Research Data در دسترس هستند. همچنین در مقاله‌ی [5] تعدادی نگاره رویداد پردازش شده‌ی عمومی برای مساله‌ی پیش‌بینی نتیجه معرفی شده است که با تعریف تابع نتیجه، برچسب خورده‌اند. به عنوان مثال مجموعه داده‌ی production مربوط به فرایند تولید و با برچسب «آیا تعداد سفارش های کاری مردود شده بیش از صفر است یا خیر» موجود است. مجموعه داده‌ی traffic_fine نیز مربوط به پلیس محلی ایتالیا و حاوی اطلاعات جرایم رانندگی است و برچسب نتیجه اینگونه تعریف می‌شود: «آیا بازپرداخت جریمه به طور کامل انجام شده یا خیر.» مجموعه داده‌ی BPIC-2012 فرایند درخواست وام و برچسب نتیجه نیز شامل دو مقدار تایید و رد است.

5. نتایج به‌دست‌آمده و پیش‌بینی‌شده

5.1 نتایج به دست‌آمده

روش‌های کلاسیک یادگیری ماشین شامل رگرسیون لجستیک، ماشین بردار پشتیبان، جنگل تصادفی و XGBoost پیاده‌سازی شده‌اند که نتایج آن در جدول 3 نشان داده شده است. همانطور که در جدول مشاهده می‌شود روش XGBoost تقریبا بهتر از بقیه عمل کرده است.

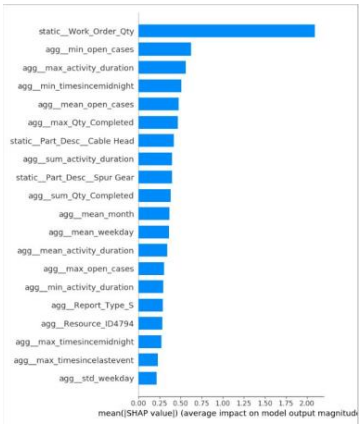
جدول 3 مقدار AUC در روش‌های کلاسیک یادگیری ماشین

Method / Dataset	LR	SVM	RF	XGBoost
production	0.67	0.66	0.68	**0.71**
traffic_fine	0.66	**0.67**	0.65	0.66

از بین روش‌های یادگیری عمیق نیز شبکه‌ی LSTM روی مجموعه داده‌ی BPIC-2012 آزمایش شده است. در پیاده‌سازی LSTM از دو لایه‌ی bidirectional استفاده کردیم و نهایتا روی این مجموعه داده به AUC برابر 0.69 به طور میانگین رسیدیم.

برای پیش‌بینی با استفاده از شباهت میان نمونه‌ها، به توابع فاصله‌ی مرسوم مانند اقلیدسی اکتفا نکرده و یک مدل یادگیری ماشین با استفاده از XGBoost برای یادگیری شباهت میان نمونه‌ها پیاده‌سازی کرده‌ایم، از این مدل فعلا برای پیش‌بینی زمان باقی‌مانده تا انتهای فرایند استفاده کرده و نتایج فعلی امیدبخش هستند.

برای تفسیرپذیری نیز از روش SHAP که با ترکیب ایده‌ی LIME و نظریه‌ی بازی‌ها و برای تعیین اهمیت ویژگی‌ها ارائه شده، استفاده کردیم. نمونه‌ای از نمودار روش SHAP روی مجموعه داده production در شکل 2 ترسیم شده است که اهمیت ویژگی‌ها را به شکل نزولی نمایش داده است.



شکل 2 روش SHAP برای مجموعه داده‌ی production

5.2 نتایج پیش‌بینی‌شده

در ادامه‌ی پژوهش، معماری‌هایی مانند ترنسفورمر را بررسی کرده و تاثیر روش‌های کدگذاری روی دقت پیش‌بینی و البته تفسیر را میسنجیم. همچنین به تکمیل ایده‌ی پیش‌بینی نتیجه براساس شباهت پرداخته و نهایتا به دنبال ترکیب روش‌های پیش‌بینی نتیجه و تفسیر آن هستیم.

چالش مهم دیگر ارزیابی تفسیر است که در حال حاضر چالش مهمی در هوش مصنوعی تفسیرپذیر و مخصوصا در فرایندکاوی است. بررسی روش‌های فعلی ارائه شده برای ارزیابی تفسیر و سنجش کارایی آن‌ها در فرایندکاوی جزو کارهای آتی این پژوهش است.

مراجع

[1] Dumas, Marlon, Marcello La Rosa, Jan Mendling, and Hajo A. Reijers. *Fundamentals of business process management,* Vol. 2. Heidelberg: Springer, 2018.

[2] Van Der Aalst, Wil, and Wil van der Aalst. *Data science in action.* Springer Berlin Heidelberg, 2016.

[3] van der Aalst, Wil MP, and Josep Carmona. *Process mining handbook.* Springer Nature, 2022.

[4] Molnar, Christoph. *Interpretable machine learning.* Lulu. com, 2020.

[5] Teinemaa, Irene, Marlon Dumas, Marcello La Rosa, and Fabrizio Maria Maggi. "*Outcome-oriented predictive process monitoring: Review and benchmark.*" ACM Transactions on Knowledge Discovery from Data (TKDD) 13, no. 2 (2019): 1-57.

[6] Stierle, Matthias, Jens Brunk, Sven Weinzierl, Sandra Zilker, Martin Matzner, and Jörg Becker. *"Bringing light into the darkness-A systematic literature review on explainable predictive business process monitoring techniques."* (2021).

[7] De Leoni, Massimiliano, Wil MP Van Der Aalst, and Marcus Dees. *"A general process mining framework for correlating, predicting and clustering dynamic behavior based on event logs."* Information Systems 56 (2016): 235-257.

[8] Di Francescomarino, Chiara, Marlon Dumas, Fabrizio Maria Maggi, and Irene Teinemaa. *"Clustering-based predictive process monitoring."* IEEE transactions on services computing 12, no. 6 (2016): 896-909.

[9] Senderovich, Arik, Chiara Di Francescomarino, Chiara Ghidini, Kerwin Jorbina, and Fabrizio Maria Maggi. *"Intra and inter-case features in predictive process monitoring: A tale of two dimensions."* In Business Process Management: 15th International Conference, BPM 2017, Barcelona, Spain, September 10–15, 2017, Proceedings 15, pp. 306-323. Springer International Publishing, 2017.

[10] Sindhgatta, Renuka, Chun Ouyang, and Catarina Moreira. *"Exploring interpretability for predictive process analytics."* In International Conference on Service-Oriented Computing, pp. 439-447. Cham: Springer International Publishing, 2020.

[11] Bukhsh, Zaharah Allah, Aaqib Saeed, Irina Stipanovic, and Andre G. Doree. *"Predictive maintenance using tree-based classification techniques: A case of railway switches."* Transportation Research Part C: Emerging Technologies 101 (2019): 35-54.

[12] Wang, Jiaojiao, Dongjin Yu, Chengfei Liu, and Xiaoxiao Sun. *"Outcome-oriented predictive process monitoring with attention-based bidirectional LSTM neural networks."* In 2019 IEEE International Conference on Web Services (ICWS), pp. 360-367. IEEE, 2019.

[13] Hinkka, Markku, Teemu Lehto, Keijo Heljanko, and Alexander Jung. *"Classifying process instances using recurrent neural networks."* In Business Process Management Workshops: BPM 2018 International Workshops, Sydney, NSW, Australia, September 9-14, 2018, Revised Papers 16, pp. 313-324. Springer International Publishing, 2019.

[14] De Weerdt, Jochen. *"Process Outcome Prediction: CNN vs. LSTM (with Attention)."* Lecture Notes in Business Information Processing 397 (2021).

[15] Pasquadibisceglie, Vincenzo, Annalisa Appice, Giovanna Castellano, Donato Malerba, and Giuseppe Modugno. *"ORANGE: outcome-oriented predictive process monitoring based on image encoding and CNNs."* IEEE Access 8 (2020): 184073-184086.

[16] Harl, Maximilian, Sven Weinzierl, Mathias Stierle, and Martin Matzner. *"Explainable predictive business process monitoring using gated graph neural networks."* Journal of Decision Systems 29, no. sup1 (2020): 312-327.

[17] Jain, Sarthak, and Byron C. Wallace. *"Attention is not explanation."* arXiv preprint arXiv:1902.10186 (2019).

[18] Liu, Xuan, Xiaoguang Wang, and Stan Matwin. *"Improving the interpretability of deep neural networks with knowledge distillation."* In 2018 IEEE International Conference on Data Mining Workshops (ICDMW), pp. 905-912. IEEE, 2018.