

پاتریوت: یک ابزار شتاب‌دهنده‌ی قوی برای استخراج اطلاعات در سکویهای اینترنت اشیا

امیر مهدی حسینی منزله^۱

^۱ استادیار، دانشکده‌ی مهندسی کامپیوتر، دانشگاه علم و صنعت ایران

تهران، ایران

monazzah@iust.ac.ir

چکیده

به لطف اینترنت اشیا تعداد دستگاه‌هایی که به اینترنت متصل می‌شوند روز به روز در حال افزایش است و انتظار می‌رود اینترنت اشیا به انقلاب صنعتی بعدی در جهان تبدیل گردد. سامانه‌های نهفته نقش اساسی در ایجاد و رشد اینترنت اشیا ایفا می‌کنند. از طرفی دیگر این شبکه اینترنت اشیا به دستگاه‌های نهفته با منابع محدود اجازه می‌دهد تا بتوانند به اینترنت متصل شوند و اطلاعات برداشت شده از محیط را با کاربران به اشتراک بگذارند. تعداد زیاد دستگاه‌های نهفته‌ی موجود در شبکه‌های IoT که به برداشت اطلاعات از محیط پیرامون خود می‌پردازند حجم زیادی اطلاعات تولید خواهد نمود که ساماندهی این اطلاعات یکی از چالش‌های اساسی تحقیقات در این حوزه در سال‌های اخیر است. در این پژوهش قصد بر آن است تا با در نظر گرفتن محدودیت‌ها و ویژگی‌های اطلاعات تولید شده در دستگاه‌های نهفته و راهکارهای ساماندهی به اطلاعات کلان‌داده‌ها در کاربردهای عادی، بستری مناسب برای انتقال و دسته‌بندی اطلاعات سامانه‌های نهفته در کاربردهای اینترنت اشیا با نام پاتریوت فراهم گردد.

کلمات کلیدی

سامانه‌های نهفته، برداشت اطلاعات، کلان‌داده‌ها، شبکه‌های اینترنت اشیا

۱- مقدمه

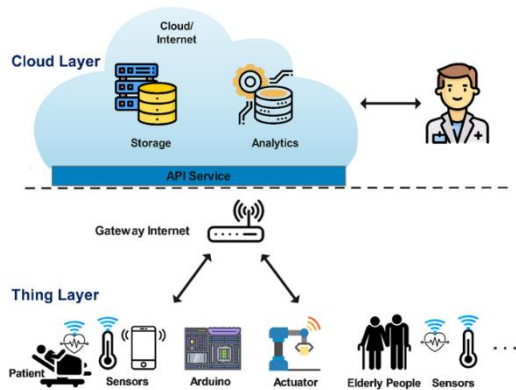
کاربردهای اینترنت اشیا شده است. این چالش که از آن با نام چالش کلان داده‌ها نیز یاد می‌شود عملاً سبب شده است تا جمع‌آوری و تحلیل این حجم از داده در کاربردهای اینترنت اشیا به یکی از اصلی‌ترین نگرانی‌های طراحان این سامانه‌ها تبدیل شود.

به منظور حل این چالش در این پژوهش قصد بر آن است تا یک سکوی برداشت و تحلیل داده به‌منظور مدیریت داده‌های تولید شده در کاربردهای اینترنت اشیا ایجاد شود. نام این سکو از واژه‌ی انگلیسی Powerful Accelerator Tools for PATRIIoT که مخفف عبارت Reteriving Information in IoT Applications است، برگرفته شده. برای طراحی این سکو نیازمند دانش طراحی Back End و دانش طراحی Front End هستیم. در بخش Front End چهارچوب پاتریوت یک محیط گرافیکی با تمرکز بر راحتی استفاده و تعامل‌پذیری مناسب با کاربران تحت وب پیاده‌سازی خواهد شد.

در بخش Back End چهارچوب پاتریوت تمامی الگوریتم‌های تحلیل و مدیریت داده‌گان پیاده‌سازی می‌شود. ابزارهای تحلیل داده در چهارچوب پاتریوت از بروزترین الگوریتم‌های ارائه شده در حوزه‌ی هوش مصنوعی همانند یادگیری تقویتی بهره می‌برند. همچنین مکانیزم‌های مختلف

امروزه سامانه‌های اینترنت اشیا، کاربردهای فراوانی در جامعه دارند. فراگیر شدن و همگرایی محاسبات ابری، کلان‌داده‌ها، تجزیه و تحلیل داده‌ها و اینترنت اشیا نحوه عملکرد و تعامل جامعه با یکدیگر را تغییر می‌دهند [۱]. اینترنت اشیا، اشیاء فیزیکی را قادر می‌سازد تا با به اشتراک گذاشتن اطلاعات و هماهنگ کردن تصمیمات، با یکدیگر در تعامل باشند. اینترنت اشیا با بهره‌برداری از فناوری‌های زیربنایی خود مانند دستگاه‌های تعبیه‌شده، فناوری‌های ارتباطی، شبکه‌های حسگر، پروتکل‌های اینترنتی و برنامه‌های کاربردی، این اشیاء را از اشیاء سنتی به هوشمند تبدیل می‌کند [۲].

برای درک بهتر همه‌گیر شدن اینترنت اشیا در جامعه شاید جالب باشد که بدانیم، در سال ۲۰۲۰ تعداد دستگاه‌های متصل به شبکه‌ی اینترنت اشیا از مرز ۲۰۰ میلیارد عبور کرده است [۳]. اغلب دستگاه‌های متصل به شبکه‌های اینترنت اشیا دستگاه‌های با منابع انرژی محدود، حافظه‌ی پایین، توان کم و فقدان بسته‌های بالا می‌باشند. وجود تعداد زیادی از این دستگاه‌ها که با فرکانس بالا به بررسی محیط پیرامون خود می‌پردازند و داده تولید می‌کنند سبب به وجود آمدن چالش جدیدی در



شکل (۱): معماری دولایه اینترنت اشیاء

اجزا در این معماری، بر اساس شبکه فیبر نوری با پهنای باند بسیار زیاد با یکدیگر ارتباط برقرار می‌کنند. یکی از محدودیت‌های قابل توجه فناوری محاسبات ابری، زمان پاسخگویی بالای خدمت^۵ است. در نتیجه، استفاده از این فناوری برای کاربردهای بی‌درنگ، به‌ویژه برای برنامه‌های حساس به تأخیر مانند سامانه‌های مراقبت سلامت، غیرممکن است. به‌همین جهت، برای برطرف کردن این چالش‌ها، محاسبات مه پیشنهاد شده است [۵].

رویکرد محاسبات مه، به‌عنوان یک لایه میانی بین دستگاه‌های اینترنت اشیاء و مراکز داده ابری عمل می‌کند. در این رویکرد، بیشتر بار پردازش داده‌ها در لایه‌ی مه توزیع می‌شود [۶]. معماری محاسبات مه شامل سه لایه است: لایه‌ی مرکز داده ابری، لایه‌ی مه^۷ و لایه اشیاء. شکل (۲) معماری سه لایه اینترنت اشیاء را نشان می‌دهد. دستگاه‌ها در لایه‌ی اینترنت اشیاء در ابتدا به لایه‌ی مه متصل می‌شوند، برخلاف فناوری محاسبات ابری که در آن کاربران به طور مستقیم با خدمت دهنده ابری ارتباط برقرار می‌کنند [۷]. در لایه‌ی مه، گرہ‌های مه که به عنوان خدمت دهنده‌های مه، خدمت دهنده‌های کوچک^۸ یا مراکز داده کوچک^۹ نیز شناخته می‌شوند، خدمات ذخیره‌سازی و پردازش داده‌ها را ارائه می‌دهند [۶]. گرہ‌ها در لایه‌ی مه شامل دستگاه‌های شبکه مانند روترها، سوئیچ‌های رایانه‌های شخصی، تلفن‌های همراه و غیره است که عملیات شبکه را انجام می‌دهند.

محاسبات مه شامل تعدادی ویژگی است که آن را از سایر چارچوب‌های محاسباتی متمایز می‌کند. برخی از ویژگی‌های آن شامل موارد زیر است [۷]:

کاهش تأخیر: از آنجایی که در معماری محاسبات مه، گرہ‌های مه در نزدیکی لایه‌ی اینترنت اشیاء قرار می‌گیرند، بنابراین پردازش داده‌ها نرخ تأخیر پایینی دارد. محاسبات مه برای کاربردهای حساس به تأخیر بسیار مناسب است.

- غیرمتمرکزسازی^{۱۰}: برخلاف محاسبات ابری، محاسبات مه دارای معماری غیرمتمرکز است. گرہ‌های مه در این معماری خدمات توزیع شده ارائه می‌دهند.

برای عیب‌یابی و کنترل فرآیند طراحی همانند همزاد دیجیتال نیز در این چهارچوب در نظر گرفته خواهد شد.

به‌منظور دریافت و ارسال پیام در چهارچوب پاتریوت در گام اول پروتکل MQTT در نظر گرفته شده است و چهارچوب پاتریوت می‌تواند از دید دستگاه‌های اینترنت اشیاء که از آن استفاده می‌کنند به عنوان یک Bocker ساده معرفی گردد، اما در پشت این Bocker یک چهارچوب تحلیل و استخراج داده نهفته است که می‌تواند از طرق مختلف به جمع‌آوری اطلاعات از دستگاه‌های اینترنت اشیاء اقدام نماید.

چهارچوب پاتریوت نه تنها می‌تواند به جمع‌آوری داده کمک کند، بلکه با تحلیل و ارائه‌ی مدل از کمیت‌های فیزیکی مورد بررسی حسگرهای اینترنت اشیاء می‌تواند داده‌های تولیدی از سمت این دستگاه‌ها را تخمین زده و این امکان را برای دستگاه‌هایی که با محدودیت انرژی مواجه هستند فراهم آورد تا نرخ نمونه‌برداری خود را کاهش دهند. در ادامه‌ی این گزارش مثالی از چگونگی کاهش انرژی مصرفی از طریق کاهش فرکانس نمونه‌برداری ارائه خواهد شد.

در ادامه‌ی این پیشنهاد پژوهشی در بخش دوم به معرفی معماری اینترنت اشیاء و یادگیری تقویتی در حوزه‌ی یادگیری ماشین خواهیم پرداخت. در بخش سوم، به بیان یک مثال عملی از میزان تأثیر کاهش نرخ نمونه‌برداری بر کاهش انرژی مصرفی می‌پردازیم. در نهایت در بخش چهارم یک جمع‌بندی و زمان‌بندی برای کارهای آتی ارائه خواهد شد.

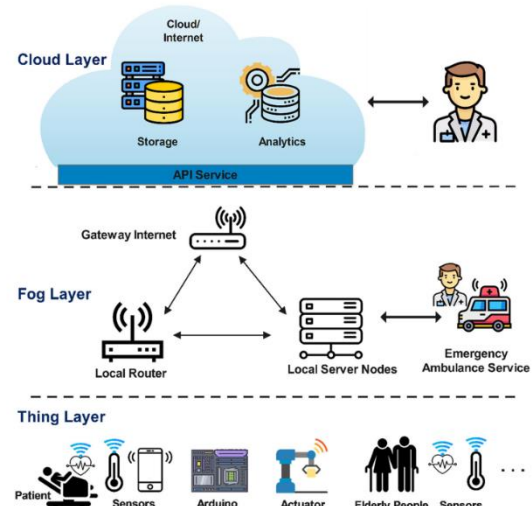
۲- پیشنهادها

در این بخش نخست به بیان کلیاتی در مورد سامانه‌های کامپیوتری و معماری اینترنت اشیاء می‌پردازیم. سپس با توجه به استفاده از مفاهیم یادگیری ماشین و یادگیری تقویتی در چهارچوب پاتریوت مقدمات مربوط به این مفاهیم ارائه خواهند شد.

۲-۱- سامانه‌های کامپیوتری و معماری اینترنت اشیاء

در سال‌های اخیر پیشرفت اینترنت اشیاء در سامانه‌های مراقبت سلامت، تأثیر خوبی بر سامانه‌های کنترل سلامت بیمار داشته است. در ابتدا، محاسبات ابری به‌عنوان یک بستر برای پردازش اطلاعات ارائه شد. فناوری محاسبات ابری امکان ارائه انواع خدمات در بستر زیرساخت به‌عنوان خدمت (IaaS)^{۱۱}، پلتفرم به‌عنوان خدمت (PaaS)^{۱۲} و نرم‌افزار به‌عنوان خدمت (SaaS)^{۱۳} را فراهم می‌کند. پایداری، قابلیت اطمینان و توانایی‌های محاسباتی قدرتمند، محاسبات ابری را به فناوری غالب دهه گذشته تبدیل کرده است [۴].

معماری یک سامانه مبتنی بر ابر عموماً از دولایه تشکیل شده است: لایه مرکز داده ابری^{۱۴} که از مجموعه‌ای از خدمت دهنده‌های ابری تشکیل شده است و لایه اشیاء که شامل دستگاه‌ها و حسگرها است. شکل (۱) معماری دولایه سامانه‌های اینترنت اشیاء را نشان می‌دهد. از جمله مزایای این معماری، می‌توان به ظرفیت ذخیره‌سازی بالا و توانایی محاسباتی با کارایی بالای خدمت دهنده‌های ابری اشاره کرد. همچنین،



شکل (۲): معماری سه لایه اینترنت اشیا [۵]

- تحرک^{۱۱}: گره‌های مه می‌توانند مستقیماً به دستگاه‌های تلفن همراه متصل شوند و بنابراین در کاربردهایی که دستگاه‌های نهایی در حرکت هستند قابل استفاده‌اند.
- قابلیت همکاری: گره‌های مه قادر به کار بر روی انواع پلتفرم‌ها با دامنه‌ها و خدمات مختلف هستند.
- برنامه‌های بی‌درنگ: محاسبات مه تعامل بی‌درنگ بین لایه‌ی اینترنت اشیا و گره‌های مه را فعال می‌کند.
- مقیاس‌پذیری: لایه‌ی مه شامل شبکه‌های حسگر با مقیاس‌های مختلف است. دستگاه‌ها در مقیاس بزرگ نیز می‌توانند با یکدیگر در ارتباط باشند.
- ناهمگونی^{۱۲}: ماهیت ناهمگونی گره‌های مه و دستگاه‌های انتهایی باعث می‌شود که بتوانند با یکدیگر روی پلتفرم‌های مختلف کار کنند.

۲-۲- یادگیری تقویتی

یادگیری از طریق تعامل^{۱۳} با محیط شاید نخستین ایده‌ای باشد که به محض روبرو شدن با واژه یادگیری به ذهن هرکسی می‌رسد. برقراری این ارتباط با محیط پیرامون، اطلاعات زیادی را در مورد علت و معلول‌ها، پیامدهای هر عمل و نیز عمل‌هایی که برای رسیدن به اهداف باید انجام داد، فراهم می‌کند [۸]. یادگیری از طریق تعامل با محیط، ایده‌ای است که تقریباً زیربنای تمام نظریه‌های یادگیری و هوشمندی به حساب می‌آید [۹].

یادگیری تقویتی^{۱۴}، به معنای یادگرفتن عمل مناسب از میان مجموعه اعمال مجاز برای یک موقعیت خاص بر اساس جایزه^{۱۵} و جریمه‌های^{۱۶} دریافتی است [۱۰]. عبارت دیگر، در یادگیری تقویتی، موضوع مورد چالش عبارت است از: چگونه نگاشت موقعیت‌ها به عمل‌ها بطوری که یک سیگنال پاداش، عددی بیشینه شود [۹]. نکته مهم این است که در این روش برخلاف اکثر روش‌های یادگیری ماشین^{۱۷}، به یادگیرنده^{۱۸} گفته نمی‌شود که در یک موقعیت خاص چه عملی را باید

انجام دهد بلکه یادگیرنده خودش باید با امتحان کردن عمل‌های مختلف در آن موقعیت، دریابد امید دریافت پاداش کدام یک از عمل‌هایش برای آن موقعیت بیشتر است [۱۰]. در اغلب مواقع، عمل‌ها ممکن است نه تنها پاداش‌های آنی، بلکه موقعیت‌های بعدی و به تبع آن تمام پاداش‌های آینده را تحت تأثیر قرار دهند. به عبارت دیگر برای دستیابی به پاداش واقعی باید سلسله‌ای از عمل‌ها انجام شود و این مفهوم، همان پاداش با تأخیر^{۱۹} است. نکته مهم در مورد مبحث یادگیری تقویتی این است که یادگیری تقویتی به عنوان یک مجموعه از الگوریتم‌های یادگیری تعریف نمی‌شود، بلکه به عنوان یک مسئله یادگیری تعویف می‌شود. هر الگوریتمی که منجر به حل این مسئله یادگیری شود، یک الگوریتم یادگیری تقویتی نامیده می‌شود [۱۱]. مسئله یادگیری تقویتی با مسئله یادگیری با ناظر^{۲۰} متفاوت است. در یادگیری با ناظر، فرآیند یادگیری بر اساس مثال‌های فراهم شده یا همان داده‌های آموزشی^{۲۱} توسط یک ناظر خارجی خبره صورت می‌گیرد. اما با وجود قابلیت بالای یادگیری با ناظر، استفاده از آن به تنهایی در کاربردهایی که داده‌های آموزشی در دسترس نیست و نیازمند یادگیری از طریق تعامل با محیط هستند، مناسب نیست. در واقع دسته‌ای از مسائل هستند که به دست آوردن مثال‌هایی از رفتار مطلوب که هم صحیح بوده و هم نماینده‌ی تمام موقعیت‌هایی باشند که عامل باید در آن‌ها عمل کند، غیرممکن یا پر هزینه است.

نکته‌ی قابل توجه دیگری که در یادگیری تقویتی وجود دارد، مصالحه بین کاوش^{۲۲} و بهره‌گیری^{۲۳} است. برای به دست آوردن پاداش زیاد، یک عامل یادگیری تقویتی باید عمل‌هایی را انتخاب کند که در گذشته امتحان کرده است و در ازای آن‌ها پاداش خوبی دریافت نموده است، اما باید دقت داشت که برای کشف چنین عمل‌هایی، لازم است عامل در ابتدا عمل‌هایی را امتحان کند که قبلاً انتخاب نکرده است. بنابراین عامل باید برای دستیابی به پاداش، از تجربیات قبلی خود استفاده کند و به علاوه برای انتخاب عمل بهتر در آینده، کاوش کند. بنابراین عامل باید در ابتدا عمل‌های متنوعی را امتحان کند و پس از به دست آوردن شناخت کافی از محیط به عمل‌هایی توجه کند که بهتر به نظر می‌رسند.

۲-۲-۱- یادگیری تقویتی در زندگی روزمره

در این بخش برخی از مثال‌هایی که در زندگی روزمره با آن‌ها مواجه هستیم را بررسی می‌کنیم تا به درک مناسب‌تری از یادگیری تقویتی برسیم.

شیری که در سیرک، پریدن از حلقه‌ی آتش را تعلیم می‌بیند. بارها با شکست رو به رو می‌شود تا آن که با پرش موفق از حلقه‌ی آتش پاداش دریافت می‌کند، بیشتر نسبت به درستی یا نادرستی حرکت اندام-هایش آگاه می‌شود. در واقع انتخاب هر یک از حرکات با توجه به طرح-ریزی و قضاوت‌های شهودی و فوری در مورد مطلوبیت مکان‌ها و حرکت-های خاص صورت می‌گیرد.

یک ربات متحرک جمع‌آوری زباله می‌تواند تصمیم بگیرد که به اتاق جدیدی وارد شده و برای جمع‌آوری زباله‌های بیشتر جستجو نماید و یا برای یافتن راه بازگشت به موقعیت شارژ باتری خود کمک کند. این ربات بر این اساس تصمیم خود را اتخاذ می‌کند که در گذشته به چه اندازه ساده و سریع توانسته است شارژر را بیابد.

یک کنترل‌گر تطبیقی^{۲۴}، پارامترهای یک عملیات پالایش نفت را در زمان واقعی تنظیم می‌کند. این کنترل‌گر مصالحه‌ای میان بازدهی^{۲۵}، کیفیت^{۲۶} و هزینه را بر اساس هزینه‌های حاشیه‌ای خاص انجام می‌دهد، بدون آن‌که در مورد رعایت نقاط پیشنهادی توسط مهندسان سخت-گیرانه عمل کند. اگر پالایش به خوبی انجام گیرد، کنترل‌گر پاداش می-گیرد و این پاداش با توجه به هدف کنترل‌گر یعنی کاهش هزینه‌ها و ارتقای کیفیت پالایش تنظیم می‌شود.

ویژگی‌هایی که نقطه مشترک تمام این مثال‌ها را تشکیل می‌دهند، تعامل یک عامل تصمیم‌گیرنده‌ی فعال با محیط اطرافش است، به طوری که عامل سعی دارد با وجود ناپیوستگی^{۲۷} موجود در محیط، به یک هدف دست یابد. عمل‌های عامل مجاز هستند در حالت آینده^{۲۸} محیط تأثیر بگذارند. در عین حال، در تمام این مثال‌ها، تأثیر عمل‌ها به طور کامل قابل پیش‌بینی نیست. بنابراین عامل باید مرتباً محیط پیرامونش را مشاهده کند و به طور مناسب عکس‌العمل نشان دهد. در تمام این مثال-ها عامل می‌تواند در طول زمان از تجربیاتش برای بهبود کارایی خود استفاده کند. عامل می‌تواند از قبل دانشی در مورد مسئله مورد بررسی داشته باشد. چنین دانشی خواه از تجربیات قبلی در مسائل به دست آمده باشد و خواه از طریق طراحی و تکامل تدریجی ایجاد شده باشد، می‌تواند آن چه را که برای یادگیری عامل راحت یا مفید است تحت تأثیر قرار دهد، اما با این وجود تعامل با محیط برای تنظیم رفتار عامل به منظور تطبیق با ویژگی‌های خاص آن مسئله، ضروری است.

۲-۲-۲- عناصر یادگیری تقویتی

علاوه بر محیط و عامل، می‌توان برای یک سیستم یادگیری تقویتی، چهار عنصر اصلی دیگر شناسایی کرد. این عناصر عبارتند از: سیاست^{۲۹}، تابع پاداش^{۳۰}، تابع ارزش^{۳۱} و در برخی از سیستم‌ها به صورت اختیاری یک مدل از محیط^{۳۲} [۹].

- **سیاست**: یک سیاست، نحوه‌ی رفتار کردن عامل یادگیرنده را در یک زمان معین تعریف می‌کند. صرف نظر از جزئیات، یک سیاست نگاشتی از حالت‌های محیط به عمل‌هایی است که در آن حالت‌ها می‌توانند انتخاب شوند. سیاست ممکن است در برخی موارد، یک تابع ساده یا یک جدول جستجو باشد و در برخی موارد، محاسباتی پر حجم همچون یک فرآیند جستجو را شامل شود. سیاست، هسته‌ی اصلی یک عامل یادگیری تقویتی است به گونه‌ای که می‌تواند به تنهایی برای تعیین رفتار عامل استفاده شود. به طور کلی، سیاست‌ها می‌توانند تصادفی باشند که در این صورت می‌توان گفت یک

سیاست، احتمال انتخاب هر عمل توسط عامل را در هر حالت نشان می‌دهد.

- **تابع پاداش**: یک تابع پاداش، هدف را در یک مسئله‌ی یادگیری تقویتی تعریف می‌کند. در واقع می‌توان گفت این تابع هر حالت^{۳۳} درک شده از محیط را به یک عدد اسکالر نگاشت^{۳۴} می‌کند. باید دقت داشت که هدف اصلی یک عامل یادگیری تقویتی، بیشینه کردن^{۳۵} مجموع پاداشی است که به دست می‌آورد. این تابع می‌تواند به عنوان مبنایی برای تغییر سیاست عمل کند. برای مثال، اگر عملی که توسط سیاست عامل انتخاب می‌شود، پاداش کمی را در پی داشته باشد، سیاست می‌تواند تغییر داده شود، به طوری که در آینده در آن موقعیت، عامل عمل دیگری را انتخاب نماید. در کل، توابع پاداش نیز می‌توانند تصادفی باشند.

- **تابع ارزش**: بر خلاف یک تابع پاداش که مشخص می‌کند چه چیزی در زمان فعلی مناسب است، یک تابع ارزش تعیین کننده‌ی آن است که در بلند مدت^{۳۶} چه چیزی سودمند است. به عبارت دیگر، ارزش یک حالت، کل میزان پاداشی را معین می‌کند که یک عامل می‌تواند انتظار داشته باشد با شروع از آن حالت، در آینده جمع‌آوری کند. بر خلاف پاداش‌ها که مطلوبیت حقیقی و لحظه‌ای حالت‌های محیط را نشان می‌دهند، ارزش‌ها مطلوبیت بلند مدت حالت‌ها را با در نظر گرفتن دنباله‌ی حالت‌ها از آن حالت تا حالت هدف و پاداش-های حاصل از آن‌ها، تعیین می‌کنند. برای مثال یک حالت ممکن است همواره پاداش لحظه‌ای کوچکی را تولید کند، اما ارزش زیادی داشته باشد، چرا که این حالت در ادامه عامل را به حالت‌هایی می‌رساند که پاداش بسیاری فراهم می‌کنند. البته عکس این موضوع هم امکان‌پذیر است. می‌توان گفت که پاداش‌ها مقادیر اولیه برای یک عامل هستند، در حالی که ارزش‌ها به عنوان پیش‌بینی‌هایی از پاداش‌ها، مقادیر کمکی برای عامل به شمار می‌آیند. دلیل این موضوع مشخص است: بدون پاداش، در واقع اصلاً ارزشی وجود نخواهد داشت و هدف اصلی تخمین ارزش‌ها، دستیابی به پاداش بیشتر است، اما با این وجود، این ارزش‌ها هستند که در زمان تصمیم‌گیری و ارزیابی^{۳۷} تصمیم‌ها بیشتر مورد توجه قرار می‌گیرند. بر اساس قضاوتی که مورد ارزش هر عمل صورت می‌گیرد، یکی از عمل‌ها انتخاب می‌شود و ما به دنبال عمل‌هایی هستیم که ما را به سمت حالت‌هایی نه با یک پاداش بیشتر بلکه با یک ارزش بیشتر هدایت می‌کنند، زیرا این عمل‌ها در بلند مدت پاداش بزرگ‌تری را برای ما به ارمغان خواهند آورد. در تصمیم‌گیری و طرح‌ریزی، ارزش، کمیتی است که بسیار مورد توجه می‌باشد. متأسفانه، تعیین ارزش‌ها بسیار دشوارتر از پاداش‌هاست زیرا پاداش‌ها مستقیماً از محیط دریافت می-

مدل‌ها برای طرح‌ریزی استفاده می‌شوند. روش‌های یادگیری تقویتی با روش‌های برنامه‌سازی پویا ارتباط نزدیکی دارند.

۳- مثال کاربردی: سامانه‌های مراقبت از سلامت

در سامانه‌های کامپیوتری با کاربرد مراقبت‌های بهداشتی، یکی از چالش‌های اساسی اجرای بی‌درنگ وظایف و افزایش قابلیت اطمینان سامانه است. همچنین محدودیت عمر باتری و زیاد بودن مصرف انرژی به دلیل نیاز به دقت بالای حسگرهای مورد استفاده در زمینه مراقبت بهداشتی، از دیگر چالش‌های موجود در این زمینه است. در این بخش قصد داریم تا به بررسی پژوهش‌های پیشین صورت گرفته در حوزه سامانه‌های مراقبت بهداشتی بپردازیم. عمده پژوهش‌های صورت گرفته در این حوزه به بهبود مصرف انرژی در حسگرهای پوشیدنی و همچنین افزایش قابلیت اطمینان سامانه‌های کامپیوتری می‌پردازد.

در پژوهش [۱۲] پژوهشگران با استفاده از الگوریتم‌های بهینه‌سازی ازدحام ذرات (MO-PSO)^{۴۳} و الگوریتم بهینه‌سازی کلونی مورچه‌ها (MO-ACO)^{۴۴} روشی پویا برای تکثیر داده‌ها ارائه کردند. در روش ارائه شده در این پژوهش، از الگوریتم PSO برای تعیین اینکه کدام فایل باید تکرار شود و چه زمانی باید تکرار شود استفاده می‌شود. همچنین از الگوریتم ACO برای تعیین محل جایگذاری داده‌ها استفاده می‌شود. در این پژوهش نحوه عملکرد به این صورت است که با افزایش تعداد نکرارها از داده‌ها و قرار گرفتن داده‌ها در موقعیت‌های مختلف و ایجاد افزونگی، قابلیت اطمینان سامانه ابری افزایش می‌یابد.

در پژوهش [۱۳] یک سامانه تشخیص درد^{۴۵} بی‌درنگ ارائه شده است. این سامانه با استفاده از حسگرها و ثبت علائم بدن و همچنین با استفاده از بینایی ماشین با در نظر گرفتن پارامترهای رفتاری بیمار، وجود درد در کاربر را تشخیص می‌دهد. در روش ارائه شده، هدف بهبود مصرف انرژی و افزایش دقت حسگرها برای نظارت طولانی مدت کاربر و افزایش قابلیت اطمینان و دسترس‌پذیری سامانه است. در ابتدا میزان باتری حسگرها بررسی می‌شود، اگر میزان باتری از سطح مشخصی کمتر باشد، محاسبات در لایه حسگر انجام می‌شود زیرا ارسال داده‌ها به لایه می و دریافت آن نیاز به مصرف انرژی بیشتری دارد. اگر میزان باتری از حد مشخص شده بیشتر باشد، میزان علائم درد کاربر بررسی می‌شود. اگر کاربر بدون حالات درد تشخیص داده شود، محاسبات در لایه حسگر انجام می‌شود زیرا نیاز به پردازش سنگین نیست. اما اگر در علائم کاربر حالت درد تشخیص داده شود، محاسبات در لایه می انجام می‌شود.

در پژوهش [۱۴] پژوهشگران یک سامانه چند هدفه نظارت بر بیمار مبتنی بر اینترنت اشیا ارائه کرده‌اند که با استفاده از روش خودآگاه ODA^{۴۶} مصرف انرژی را مدیریت می‌کند و قابلیت اطمینان سامانه را افزایش می‌دهد. این سامانه وضعیت سلامت کاربران، فعالیت و سطح انرژی فعلی گره حسگر را مشاهده می‌کند و با توجه به مشاهدات، اهداف تعریف شده را اولویت‌بندی می‌کند. در این پژوهش چهار واحد برای پردازش و تصمیم‌گیری تعریف شده است. واحد اول، واحد مشاهده است

شوند، اما ارزش‌ها باید به صورت تدریجی با استفاده از دنباله مشاهداتی که عامل در طول عمرش به دست می‌آورد، تخمین زده شوند. در حقیقت، نقش اصلی تخمین ارزش تقریباً مهمترین موضوعی است که در چند دهه اخیر پیرامون یادگیری تقویتی شناخته شده است. روش‌های یادگیری تقویتی دیگری هم وجود دارند که از مفهوم تابع ارزش استفاده نمی‌کنند و در نتیجه در این روش‌ها تخمین این تابع نیز مورد نیاز است.

به طور کلی دو گروه از روش‌ها برای حل مسائل یادگیری تقویتی موجود است. گروه اول، شامل جستجوی مستقیم در فضای سیاست تا زمان دستیابی به سیاستی است که در محیط به خوبی عمل کند. برای مثال، روش‌های مبتنی بر جستجو همچون الگوریتم‌های ژنتیک^{۴۸}، شبیه‌سازی تبرید^{۴۹} و سایر روش‌های بهینه‌سازی تابع را می‌توان برای حل مسائل یادگیری تقویتی به کار برد. این روش‌ها مستقیماً در فضای سیاست‌ها جستجو می‌کنند، بدون آنکه حتی توابع ارزش سیاست را بررسی نمایند. به همین دلیل به آن‌ها روش‌های جستجوی سیاست گفته می‌شود. اگر فضای سیاست‌ها به قدر کافی کوچک باشد و یا بتواند به گونه‌ای ساختاردهی شود که سیاست‌های خوب به سادگی در آن پیدا شوند، آن‌گاه روش‌های جستجوی سیاست می‌توانند موثر واقع گردند. به علاوه، روش‌های جستجوی سیاست در مسائلی که عامل قادر نیست به درستی و با دقت حالت محیط پیرامونش را حس کند، مفید هستند.

گروه دوم از روش‌های آماری و راهکارهای برنامه‌سازی پویا^{۴۰} استفاده می‌کنند و میزان سودمندی انتخاب هر یک از عمل‌های ممکن را در حالت‌های محیط تخمین می‌زنند. این گروه از مزایای ساختار خاص مسائل یادگیری تقویتی بهره‌برداری می‌کنند، ویژگی‌هایی که در مسائل بهینه‌سازی^{۴۱} در حالت کلی در دسترس نیستند. در حقیقت، منظور از یادگیری تقویتی، این دسته از یادگیری است که در تعامل با محیط حاصل می‌شود، چیزی که روش‌های جستجوی سیاست قادر به انجام آن نیستند. ما بر این باوریم که روش‌هایی با قابلیت بهره‌گیری از جزییات مربوط به تعاملات رفتاری انفرادی در بسیاری از موارد کارا تر از روش‌های جستجوی سیاست هستند. روش‌های جستجوی سیاست بخش عمده‌ای از ساختار سودمند مسئله یادگیری تقویتی را نادیده می‌گیرند. این روش‌ها این حقیقت رو نادیده می‌گیرند که سیاستی که دنبال آن هستند، تابعی از حالت‌ها به عمل‌ها است و به این مسئله توجه ندارند که عامل، از کدام حالت‌ها در طول حیات خود می‌گذرد، یا کدام عمل‌ها را انتخاب می‌کند. این اطلاعات می‌تواند گمراه‌کننده باشند.

- **مدل محیط** : یک مدل، رفتار محیط را تقلید^{۴۲} می‌کند. برای مثال، با در نظر گرفتن یک حالت و عمل معین، مدل ممکن است حالت و پاداش بعدی را پیش‌بینی کند. باید دقت کرد که در یادگیری تقویتی، روابط دینامیکی و فیزیکی محیط به صورت فرمول‌هایی بسته از مدل محیط مورد نیاز نیست، بلکه

وضعیت سلامت کاربر دچار مشکل نباشد، گره‌های حسگر بر روی حالت با دقت کمتر تنظیم می‌شوند و انرژی کمتری مصرف می‌کنند. مصرف انرژی کمتر منجر به طولانی‌تر شدن زمان در دسترس بودن حسگرها می‌شود و این امر موجب افزایش قابلیت اطمینان و دسترس‌پذیری بالاتر حسگرها می‌شود.

۴- مشاهدات

در این پژوهش، به بررسی یک روش نمونه‌برداری ضربان قلب می‌پردازیم تا نرخ نمونه‌برداری بهینه دستگاه را مشخص کرده و نشان دهیم مصرف انرژی دستگاه پوشیدنی را می‌توان با نرخ نمونه‌برداری بهینه کاهش و قابلیت اطمینان و دسترس‌پذیری دستگاه را افزایش داد. در این پژوهش، برای نمونه‌برداری ضربان قلب، از مچ‌بند هوشمند شیائومی^{۵۱} استفاده شده است. این مچ‌بند دارای حسگرهای ژيروسکوپ^{۵۲}، شتاب‌سنج^{۵۳} و شمارنده ضربان قلب است که در این پژوهش از حسگر شتاب‌سنج برای سنجش نوع فعالیت کاربر و از حسگر شمارنده ضربان قلب، برای شمارش تعداد ضربان قلب استفاده شده است. همچنین ظرفیت باتری این مچ‌بند ۱۳۵mAh است و مدت‌زمان کارکرد این مچ‌بند با ظرفیت باتری ۱۰۰ درصد، ۲۰ روز است. در این پژوهش، نمونه‌برداری فعالیت‌های یک کاربر طی مدت‌زمان ۲۴ ساعت بررسی شده است که فعالیت‌ها به سه دسته خوابیدن، راه رفتن و نشستن تقسیم‌بندی شده است. فرکانس استاندارد نمونه‌برداری ضربان قلب به‌صورت هر یک دقیقه یکبار تنظیم شده است و با استفاده از حسگر شتاب‌سنج، میزان راه رفتن و زمان خوابیدن کاربر به‌دست‌آمده است.

در روش ارائه شده، یک مقدار با عنوان محدوده‌ی انحراف قابل قبول در نظر می‌گیریم که این عدد مشخص‌کننده بازه‌ی عددی ضربان قلب است. به‌عنوان مثال، در صورتی که محدوده‌ی انحراف قابل قبول برابر با مقدار ۵ باشد، مقادیر ضربان قلبی که به میزان ۵ واحد کوچک‌تر و یا بزرگ‌تر از مقدار فعلی هستند، در یک بازه قرار می‌گیرند. این بازه مشخص می‌کند که ما برای نمونه‌برداری داده‌ها به‌جای زمان یک دقیقه در این بازه، تا چند دقیقه می‌توانیم صبر کنیم و نمونه‌برداری را متوقف کنیم. در صورتی که یک داده اختلافی بیش از محدوده‌ی انحراف قابل قبول از شروع بازه داشته باشد، آن را به‌عنوان داده غیرقابل حذف در نظر می‌گیریم و بازه‌ی قابل تجمیع داده‌ها قبل از آن به پایان خواهد رسید. داده غیرقابل حذف، داده‌ای است که باید حتماً در زمان مشخص شده نمونه‌برداری شود.

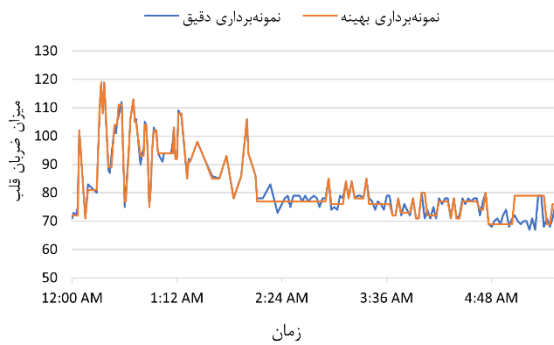
به‌عنوان مثال، مطابق جدول (۴-۱)، اعداد ۷۱ تا ۷۵ با نرخ محدوده‌ی انحراف قابل قبول ± 5 در یک بازه قرار می‌گیرند و میانگین آنها به‌عنوان نماینده این بازه انتخاب می‌شود. یعنی این اعداد به‌عنوان نمونه‌های قابل حذف در نظر گرفته می‌شوند. اما عدد ۸۱ چون با مقدار شروع بازه داده‌های قابل تجمیع (۷۱) به میزان ۵ واحد اختلاف ندارد، به‌عنوان نمونه غیرقابل حذف و دقیق در نظر گرفته می‌شود. لذا نمونه‌ی ۸۱ جلوی گسترش بازه‌ی قابل تجمیع را گرفت. حال همین عملیات

که در لایه‌ی حسگر قرار دارد و اطلاعات کاربر را مشاهده می‌کند و به لایه‌ی مه ارسال می‌کند. واحد دوم واحد تشخیص وضعیت است که وضعیت سلامت و سطح فعالیت کاربر را بررسی می‌کند. نتایج محاسبات این واحد به واحد ارزیابی اولویت ارسال می‌شود. این واحد با استفاده از سلسله مراتب اولویت‌های تعیین شده، بهترین چارچوب را برای پیکربندی حسگر انتخاب می‌کند. برای مثال اگر کاربر در وضعیت خوبی نباشد سامانه یک پیام هشدار ارسال می‌کند و تمام منابع موجود را به نظارت مستمر و دقیق اختصاص می‌دهد تا خدمات اورژانسی به کاربر برسد. در نهایت، اطلاعات به واحد اجرای سیاست ارسال می‌شود و این واحد براساس اولویت در نظر گرفته‌شده، پیکربندی دستگاه را تغییر می‌دهد. به این ترتیب، مصرف انرژی حسگر براساس سیاست تعیین شده تغییر می‌کند و کاهش مصرف انرژی، منجر به دسترس‌پذیری بالاتر و افزایش قابلیت اطمینان می‌شود.

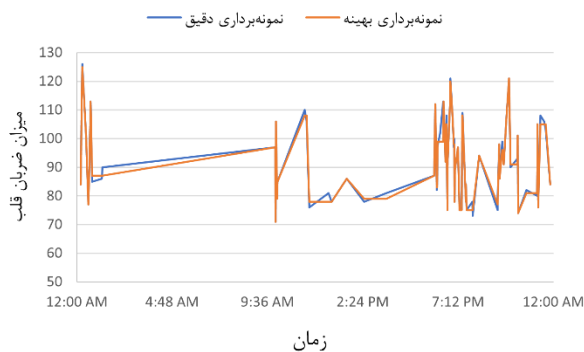
در پژوهش [۱۵] دو روش برای بهبود مصرف انرژی حسگر و طولانی‌تر شدن عمر باتری و افزایش قابلیت اطمینان دستگاه‌ها ارائه شده است. روش اول روش نزدیک بینی^{۴۷} است، در این روش قدرت و دقت حسگرها تنها براساس فعالیت فعلی کاربر تنظیم می‌شود. روش ارائه شده دوم، روش MDP^{۴۸} است. در این روش قدرت و دقت حسگرها براساس یک مدل آماری از فعالیت‌های کاربر و پویایی وضعیت باتری تنظیم می‌شود. سپس، براساس روش انتخاب شده، تنظیمات حسگرها بازپیکربندی می‌شود و مصرف انرژی حسگرها بهینه می‌شود.

در پژوهش [۱۶] یک معماری سلسله مراتبی با نام $^{49}\text{HiCH}$ معرفی شده است. این معماری یک روش مدیریت سامانه که با شرایط بیمار تنظیم شده است را به‌کار می‌گیرد که این شرایط براساس پارامترهای پزشکی بیمار، فعالیت‌ها و محیط اطراف تعریف می‌شوند. عملکرد این سامانه به این گونه است که داده‌ها از حسگرها جمع‌آوری می‌شوند، سپس در لایه‌ی مه، برنامه تصمیم‌گیری محلی که وضعیت بیمار را مشخص می‌کند، فعال می‌شود. سپس، با تحلیل داده‌های جمع‌آوری شده از حسگرها، تصمیم‌گیری انجام می‌شود و در نهایت مدیر سامانه به‌صورت محلی پیکربندی سامانه را مدیریت می‌کند. همچنین یک واحد تحلیل در ابر قرار دارد که کار پردازش و مدل‌سازی شرایط نظارتی پیچیده را بر عهده دارد.

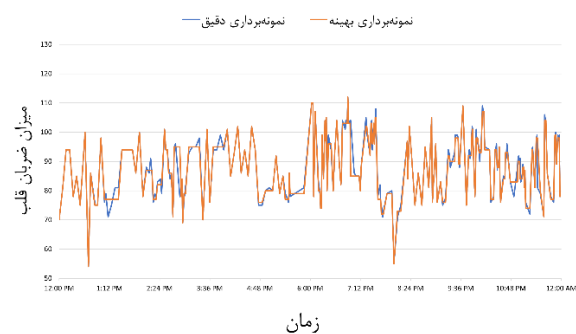
در پژوهش [۱۷] راه حل پیشنهاد شده یک کنترل‌کننده هوشمند بی‌درنگ با تأخیر کم است که در لایه‌ی مه پیاده‌سازی شده‌است. این سامانه ناهنجاری‌ها را در وضعیت بیمار تشخیص می‌دهد و براساس پارامترهای دریافت شده از حسگر و وضعیت سلامت بیمار، یک گره حسگر قابل تنظیم مجدد را تنظیم می‌کند. وضعیت سلامت بیمار، با استفاده از روش EWS^{۵۰} بررسی می‌شود. روش EWS یک روش امتیاز دهی خطر است که برای تشخیص زود هنگام وخامت سلامتی با استفاده از علائم حیاتی بیمار طراحی شده است. در این پژوهش، اگر سلامت کاربر در وضعیت وخیم باشد، گره‌های حسگر با دقت بالا بر شرایط بیمار نظارت می‌کنند تا خدمات اورژانسی به کاربر برسد و در حالتی که



شکل (۳): نمودار نمونه برداری ضربان قلب در حالت خواب



شکل (۴): نمودار نمونه برداری ضربان قلب در حالت راه رفتن



شکل (۵): نمودار نمونه برداری ضربان قلب در حالت نشستن

۵- کارهای آتی

باتوجه به گسترش روزافزون اینترنت اشیاء در کاربردهای مختلف از جمله مراقبت سلامت و وجود چالش‌ها و محدودیت‌های گوناگون در این سامانه‌ها از جمله محدودیت منابع انرژی و نیاز این سامانه‌ها به قابلیت اطمینان و دسترس پذیری بالا، در این پژوهش یک چهارچوب قدرتمند برای جمع‌آوری اطلاعات در کاربردهای اینترنت اشیاء به نام پاتریوت ارائه شد. به عنوان یکی از مزایای استفاده از چهارچوب پاتریوت اشاره شد که می‌توان از الگوریتم‌هایی برای کاهش نرخ نمونه برداری و استفاده از تخمین تحت روش‌های یادگیری ماشین استفاده کرد. برای مشاهده کاربرد این موضوع در حوزه سلامت ابتدا روش‌های گوناگون کاهش مصرف انرژی و تأثیر آن در افزایش دسترس پذیری و قابلیت اطمینان سامانه‌های مراقبت سلامت را بررسی کردیم و در ادامه، یک روش نمونه برداری بهینه از داده‌های ضربان قلب را در سه حالت خوابیدن،

جدول (۱): داده‌های ضربان قلب جمع‌آوری شده

date	time	heartRate	status
۲۰۲۲-۱۰-۱۰	۰۰:۰۰:۰۰	۷۱	sleep
۲۰۲۲-۱۰-۱۰	۰۰:۰۱:۰۰	۷۳	sleep
۲۰۲۲-۱۰-۱۰	۰۰:۰۲:۰۰	۷۲	sleep
۲۰۲۲-۱۰-۱۰	۰۰:۰۳:۰۰	۷۳	sleep
۲۰۲۲-۱۰-۱۰	۰۰:۰۴:۰۰	۷۵	sleep
۲۰۲۲-۱۰-۱۰	۰۰:۰۵:۰۰	۸۱	sleep
۲۰۲۲-۱۰-۱۰	۰۰:۰۶:۰۰	۷۵	sleep

برای جستجوی بازه‌ی قابل جمع‌آوری با شروع از نمونه‌ی با مقدار ۸۱ و حرکت به سمت نمونه‌های بعدی تکرار می‌شود. در نهایت، یک آرایه از نمونه‌برداری‌های انجام شده به دست می‌آید که این آرایه شامل یکسری از داده‌های قابل حذف و داده‌های غیرقابل حذف هستند. میزان نرخ خطا پس از ادغام داده‌های نمونه‌برداری را می‌توان از طریق فرمول (۳) محاسبه نمود:

$$Error\ Rate = \pm \frac{Acceptable\ variance}{MaxHeartRate - MinHeartRate} \quad (۱)$$

در این فرمول، نسبت محدوده‌ی انحراف قابل قبول برای در نظر گرفتن بازه اعداد را به اختلاف بیشترین میزان ضربان قلب ثبت شده به کمترین میزان ضربان قلب به دست می‌آوریم.

در ادامه، شکل (۳) نمودار نمونه‌برداری داده‌های ضربان قلب در حالت خوابیدن، شکل (۴) نمونه‌برداری داده‌های ضربان قلب در حالت راه رفتن و شکل (۵) نمونه‌برداری داده‌ها در حالت نشستن را نشان می‌دهد. خطوط آبی رنگ نمایانگر کل نمونه‌های واقعی هستند که در بازه زمانی یک دقیقه یکبار، اندازه‌گیری شده است. خطوط نارنجی رنگ نمایانگر نمونه داده‌های دقیق و غیرقابل حذف است که اگر در این زمان‌ها نمونه‌برداری انجام شود، میزان نمونه‌برداری داده‌ها بهینه خواهد بود.

در نهایت، با تقسیم طول کل بازه زمانی نمونه‌برداری داده‌ها بر تعداد نمونه‌های غیرقابل حذف، فرکانس نمونه‌برداری بهینه در هر یک از سه حالت در نظر گرفته شده برای کاربر به دست می‌آید. فرمول (۴)، نحوه محاسبه فرکانس نمونه‌برداری بهینه را نشان می‌دهد:

$$T_{Optimal} = \frac{Total-Samples}{Non-Removable_Samples} \times T_{Full-Accurate} \quad (۲)$$

$T_{Full-Accurate}$ نمایانگر فرکانس نمونه‌برداری فعلی (همان یک دقیقه یکبار) است.

در این پژوهش، فرکانس نمونه‌برداری بهینه با محدوده‌ی انحراف قابل قبول ± 5 برای حالت خوابیده برابر با مقدار ۲، برای حالت راه رفتن برابر با مقدار ۱ و برای حالت نشستن برابر با مقدار ۲ دقیقه است. یعنی در حالت خوابیده، می‌توان به جای هر یک دقیقه نمونه‌برداری، هر دو دقیقه یکبار از داده‌ها نمونه‌برداری کرد. در مثال ذکر شده، با فرکانس به دست آمده، در زمان خوابیدن ۵۰ درصد ذخیره انرژی و در زمان نشستن نیز ۵۰ درصد انرژی مصرفی را ذخیره می‌کنیم. به این ترتیب، میزان انرژی مصرفی در دستگاه پوشیدنی کاهش می‌یابد و کاهش مصرف انرژی، منجر به طولانی‌تر شدن مدت زمان کارکرد می‌شود که این امر موجب افزایش دسترس پذیری و قابلیت اطمینان دستگاه می‌شود.

جدول (۱-۵): زمان بندی پیش برد پروژه

مطالعه‌ی روش‌های پیاده‌سازی چهارچوب اینترنت اشیا	اسفند	بهمن	دی	آذر	آبان	مهر	شهریور	مرداد	تیر	خرداد	اردیبهشت	فروردین
مطالعه‌ی روش‌های یادگیری ماشین در سخت‌افزار												
پیاده‌سازی اولیه‌ی روش‌های یادگیری ماشین												
نگارش مقاله‌ی مروری بر پژوهش‌های پیشین												
پیاده‌سازی نهایی چهارچوب باتریوت												
تعیین سناریوهای و مجموعه ابزارهای مورد نظر												
پیاده‌سازی و اشکال زدایی انتهایی												
نگارش مقاله												

سال ۱۴۰۱

Energy in Fog-Enabled IoT Networks," IEEE Internet of Things Journal, vol. 4, pp. 377-382, 2017.

- [8] D. Chapman et al., "Input generalization in delayed reinforcement learning: An algorithm and performance comparisons." In Proceedings of the Twelfth International Conference on Artificial Intelligence, (1991).
- [9] R. S. Sutton et al., "Reinforcement Learning: An Introduction," The MIT Press Cambridge, (2012).
- [10] A. G. Barto et al., "Learning to act using real-time dynamic programming" Artificial Intelligence, (1995).
- [11] S. Singh et al., "Convergence Results for Single-Step On-Policy Reinforcement-Learning Algorithms." Machine Learning, (2000).
- [12] Ahmed Awad, Rashed Salem, Hatem Abdelkader, Mustafa Abdul Salam, "A Novel Intelligent Approach for Dynamic Data Replication in Cloud Environment," IEEE Access, vol. 9, pp. 4044--4054, 2021.
- [13] Emad Kasaeyan Naeini, Sina Shahhosseini, Ajan Subramanian, Tingjue Yin, Amir M Rahmani, Nikil Dutt,, "An edge-assisted and smart system for real-time pain monitoring," in 2019 IEEE/ACM International Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE), Arlington, VA, USA, 2019.
- [14] Arman Anzanpour, Humayun Rashidba, Amir M. Rahmani, Axel Jantsch, Nikil Dutt, Pasi Liljeberg, "Energy-efficient and Reliable Wearable Internet-of-Things through Fog-Assisted Dynamic Goal Management," Procedia Computer Science, vol. 101, p. 493--500, 2019.
- [15] Delaram Amiri, Arman Anzanpour, Iman Azimi, Amir M Rahmani, Pasi Liljeberg, Nikil Dutt, Marco Levorato,, "Optimizing Energy Efficiency of Wearable Sensors Using Fog-assisted Control," Fog Computing: Theory and Practice, pp. 245--268, 2020.
- [16] Iman Azimi, Arman Anzanpour, Amir M Rahmani, Tapio Pahikkala, Marco Levorato, Pasi Liljeberg, Nikil Dutt, "HiCH: Hierarchical fog-assisted computing architecture for healthcare IoT," ACM Transactions on Embedded Computing Systems (TECS), vol. 16, pp. 1--20, 2017.
- [17] Arman Anzanpour, Delaram Amiri, Iman Azimi, Marco Levorato, Nikil Dutt, Pasi Liljeberg, Amir M Rahmani, "Edge-assisted control for healthcare internet of things: A

نشستن و راه رفتن بررسی کردیم. نتایج این بررسی نشان داد با تغییر نرخ نمونه برداری، میزان انرژی مصرفی دستگاه کاهش یافته و در نتیجه دسترس پذیری و قابلیت اطمینان دستگاه افزایش پیدا می کند. در ادامه در جدول (۵-۱) نمودار زمانی پیشرفت پژوهش در نظر گرفته شده مشاهده می شود.

مراجع

- [1] Da Silva Lisboa Tigre, Matheus Felipe Ferreira and Santos, Guto Leoni and Lynn, Theo and Sadok, Djamel and Kelner, Judith and Endo, Patricia Takako, "Modeling the availability of an e-health system integrated with edge, fog and cloud infrastructures," in 2018 IEEE Symposium on Computers and Communications (ISCC), Natal, Brazil, 2018.
- [2] Al-Fuqaha, Ala and Guizani, Mohsen and Mohammadi, Mehdi and Aledhari, Mohammed and Ayyash, Moussa, "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications," IEEE Communications Surveys & Tutorials, vol. 17, pp. 2345-2376, 2015.
- [3] Razzaque, Mohammad Abdur, et al. "Middleware for internet of things: a survey." IEEE Internet of things journal 3,1 (2015): 70-90.
- [4] Kai Fan, Shanshan Zhu, Kuan Zhang, Hui Li, Yintang Yang,, "A lightweight authentication scheme for cloud-based RFID healthcare systems," IEEE Network, vol. 33, pp. 44-49, 2019.
- [5] Vu Khanh Quy, Nguyen Van Hau, Dang Van Anh, Le Anh Ngoc, "Smart healthcare IoT applications based on fog computing: architecture, applications and challenges," Complex and Intelligent Systems, vol. 1, pp. 280-281, 2022.
- [6] Arman Anzanpour, Humayun Rashidba, Amir M. Rahmani, Axel Jantsch, Nikil Dutt, Pasi Liljeberg, "Energy-efficient and Reliable Wearable Internet-of-Things through Fog-Assisted Dynamic Goal Management," Procedia Computer Science, vol. 151, p. 493-500, 2019.
- [7] Penghao Cai, Fuqian Yang, Jianjia Wang, Xing Wu, Yang Yang, Xiliang Luo. "JOTE: Joint Offloading of Tasks and

۲۸ Future State	۱ Infrastructure As A Service
۲۹ Policy	۲ Platform As A Service
۳۰ Reward Function	۳ Software As A Service
۳۱ Value Function	۴ Cloud Datacenter Layer
۳۲ Environment	۵ Server
۳۳ State	۶ Service
۳۴ Map	۷ Fog Layer
۳۵ Maximize	۸ Micro Server
۳۶ Long Term	۹ Micro Datacenters
۳۷ Evaluation	۱۰ De centralization
۳۸ Genetic Algorithm	۱۱ Mobility
۳۹ Simulated Annealing	۱۲ Heterogeneity
۴۰ Dynamic Programming	۱۳ Interaction
۴۱ Optimization	۱۴ Reinforcement Learning
۴۲ Mimic	۱۵ Reward
۴۳ Multi-Objective Particle Swarm Optimization	۱۶ Punishment
۴۴ Multi-Objective Ant Colony Optimization	۱۷ Machine Learning
۴۵ Pain Detection System	۱۸ Agent
۴۶ Observe-Decide-Act	۱۹ Delayed Reward
۴۷ Myopic Strategy	۲۰ Supervised Learning
۴۸ Marcov Decision Processes	۲۱ Training Data
۴۹ Hierarchical Computing Architecture For Healthcare	۲۲ Exploration
۵۰ Early Warning Scores	۲۳ Exploitation
۵۱ Xiaomi Mi Band ۴	۲۴ Adaptive Controller
۵۲ Gyroscope	۲۵ Throughput
۵۳ Accelerometer	۲۶ Quality
	۲۷ Uncertainty