



ارائه شده توسط:

سایت ترجمه فا

مرجع جدیدترین مقالات ترجمه شده

از نشریات معتبر

تکنیک‌های سنسور (حسگر) جستجو جهت دریافت (سنجش) به عنوان یک

معماری سرویس در اینترنت اشیا

چکیده:

اینترنت چیزها (IoT) بخشی از اینترنت آینده است و شامل میلیاردها "چیز" ارتباطی هوشمند یا اشیاء متصل به اینترنت (ICOs) است که قابلیت سنجش، فعال شدن و پردازش اطلاعات را دارند. هر ICO یک یا چند سنسور جاسازی شده دارد که بطور بالقوه مقدار زیادی از داده‌ها را حفظ می‌کند. سنسورها و سیر داده‌های مربوطه را می‌توان به طور فیزیکی و یا بطور عملی خوشه بندی کرد، که منجر به ایجاد چالش جستجو و انتخاب سنسورهای مناسب برای پرس و جو با یک راه کارآمد و موثر گردد. این مقاله پیشنهاد می‌کند که جستجوی حسگرمتنی - آگاهانه، انتخاب و رتبه بندی مدلی به نام CASSARAM را پیشنهاد می‌دهد که چالش انتخاب مناسب در واقع زیرمجموعه‌ای از سنسورهای مربوطه یک مجموعه وسیع سنسورها را با قابلیت‌ها و عملکردهای مشابه مطرح می‌کند. CASSARAM مزایای کاربری و گستره وسیعی از ویژگی‌های سنسور مانند قابلیت اطمینان، دقت، موقعیت مکانی، عمر باتری و بسیاری موارد دیگر را در نظر می‌گیرد. این مقاله اهمیت جستجوی حسگر، انتخاب و رتبه بندی برای IoT را برجسته می‌کند که ویژگی‌های مهمی از هر دو حسگر و پردازش‌های ضبط داده را شناسایی می‌کند و همچنین بحث می‌کند که چگونه استدلال معنایی و کمی را می‌توان با هم ترکیب کرد. این مقاله همچنین به چالش‌هایی مانند جستجوی حسگر مناسب توزیع شده و عبارت رابطه‌ای مبتنی بر فیلترینگ اشاره می‌کند. تست‌های CASSARAM و نتایج ارزیابی عملکرد ارائه شده و مورد بحث قرار می‌گیرند.

کلمات کلیدی: شاخص شرایط سنسورها، جستجو و انتخاب، شاخص گذاری و رتبه بندی، پرس و جو معنایی، استدلال

کمی، تلفیق داده‌های چند بعدی.

1. مقدمه

تعداد سنسورهای گسترش یافته در سرتاسر جهان، با سرعت در حال افزایش است. این سنسور ها به طور مداوم مقدار زیادی از اطلاعات را تولید می کنند. با این وجود، جمع آوری داده ها از تمام حسگرهای موجود ارزش افزوده ایجاد نمی کند، مگر اینکه آنها قادر به ارائه بینش ارزشمند باشند که در نهایت به رسیدگی به چالش هایی که ما هر روز با آن مواجه هستیم کمک کنند (مثلا مدیریت آلودگی محیط زیست و مدیریت تراکم حمل و نقل). علاوه بر این، به لحاظ مقیاس بزرگ آن، محدودیت منابع و عوامل هزینه نیز امکان پذیر نیست. هنگامی که تعداد زیادی از سنسورها با توجه به آن انتخاب استفاده می شود، در نتیجه انتخاب سنسور مناسبی¹ که به کاربران کمک می کند تا مشکلات خود را حل کنند، یک چالش و یک کار وقت گیر خواهد بود.

انتظار می رود که سنجش به عنوان مدل سرویس [1] است که در بالای زیرساخت های IoT و خدمات ساخته شود. همچنین پیش بینی می کند که سنسورها برای استفاده در اینترنت یا به صورت رایگان با پرداخت هزینه از طریق راه حل های نرم افزار در دسترس هستند. در حال حاضر، چندین راهکار میان افزاری که انتظار می رود این مدل را تسهیل کنند، در حال توسعه هستند. [2] OpenIoT، [3] GSN و xively (xively.com) بعضی از نمونه ها هستند. این راه حل های میان افزار بر روی اتصال دستگاه های سنسور به سیستم های نرم افزاری و قابلیت های مرتبط تمرکز دارد [2]. با این حال، زمانی که سنسورهای بیشتری به اینترنت متصل می شوند، قابلیت جستجو بسیار مهم می شود. این مقاله به مسئله ذکر شده در بالا اشاره می کند، زیرا ما عدم تمرکز بر انتخاب حسگر و جستجو در راه حل ها و تحقیقات موجود در IoT را بررسی می کنیم. از آنجائیکه متون مبتنی بر روش های تحقیقی نمی تواند ویژگی های بحرانی حسگر دقیق را ضبط یا حفظ کند، رویکرد سنتی جستجوی وب در انتخاب جستجوی حسگر IoT و دامنه جستجو موثر واقع نخواهد شد. یکی دیگر از رویکردهایی که می تواند دنبال شود حاشیه نویسی ابر داده (metadata) است. حتی اگر ما ابر داده های روی سنسورها (به عنوان مثال ذخیره شده در یک ذخیره سازی حسگر) یا در ابر را حفظ کنیم، قابلیت همکاری مسئله مهمی خواهد بود. علاوه بر این، مطالعه کاربری که توسط بورینگ و همکاران انجام

¹ ما اصطلاح مناسب در بخش سوم را توصیف می کنیم

شده است، [4] توضیح داده شده است که چگونه از 20 شرکت کننده خواسته شد که از طریق یک رابط کاربری ساده ابر داده ها را برای یک سنسور ایستگاه هواشناسی وارد کنند. این 20 نفر در مجموع 45 کار اشتباه را انجام دادند. الزام ورود دوباره به ابر داده در مکان های مختلف (به عنوان مثال وارد کردن ابر داده ها در GSN یک مرتبه و وارد کردن دوباره ابر داده در OpenIoT، و غیره) هنگامی رخ می دهد که ما توصیفات مشترکی نداریم. گروه انوکباتور W3C به تازگی، گزارش نهایی XG شبکه حسگر معنایی را منتشر کرده است، که یک هستی شناسی SSN را تعریف می کند [5]. هستی شناسی SSN اجازه می دهد تا توصیف سنسورها، از جمله ویژگی های آنها امکان پذیر گردد. این تلاش باعث افزایش قابلیت همکاری و دقت ناشی از عدم ورود اطلاعات دستی می شود. علاوه بر این، می توان با اجازه دادن به تولید سخت افزارهای سنسور و ارائه توصیف های حسگر با استفاده از هستی شناسی ها از چنین اشتباهاتی جلوگیری کرد، به طوری که توسعه دهندگان راه حل های IoT بتوانند در سیستم نرم افزاری خود (مثلا نقشه گذاری) آنها را بازیابی و ترکیب کنند.

بر اساس استدلال بالا، توصیف سنسور مبتنی بر هستی شناسی و مدل سازی داده ها برای راه حل های IOT مفید است. این روش همچنین پرس و جو معنایی را امکان پذیر می سازد. راه حل پیشنهادی ما اجازه می دهد تا کاربران اولویت خود را از نظر ویژگی های سنسور بیان کنند و حسگرهای مناسب را جستجو و انتخاب کنند. در مدل ما، هر دو روش استدلال کمی و تکنیک های پرس و جو معنایی جهت افزایش عملکرد سیستم با استفاده از نقاط قوت هر دو تکنیک استفاده می شود.

بقیه این مقاله به شرح زیر ساختار یافته است: در بخش دوم، ما بطور خلاصه به بررسی ادبیات و ارائه برخی از توصیفات راه حل های میان افزار پیشرو IoT و قابلیت جستجو حسگر آنها می پردازیم. ما بعداً در اینجا، معانی مشکلات و انگیزه ها در قسمت سوم را ارائه می دهیم. راه حل پیشنهادی ما، CASSARAM، با جزئیات در قسمت چهارم ارائه شده است. مدل های داده، چارچوب زمینه، الگوریتم ها و معماری ها در این بخش مورد بحث قرار می گیرند. تکنیک هایی که ما برای بهبود CASSARAM توسعه دادیم در بخش پنجم ارائه شده است. در بخش ششم، ما جزئیات اجرایی، از جمله ابزارها، پلت فرم های نرم افزاری، پلتفرم های سخت افزاری و مجموعه داده های مورد

استفاده در این تحقیق را ارائه می دهیم. ارزیابی و بحث در مورد نتایج تحقیق در بخش هفتم ارائه شده است. در نهایت، ما نتیجه و چشم انداز برای تحقیقات آینده را در بخش هشتم ارائه می کنیم.

2. پیشینه و مطالعات مرتبط

در حالت ایده آل، راه حل های میان افزار IoT باید به کاربران اجازه دهند تا آنچه را که می خواهند بیان کنند و بدون اینکه از کاربران بخواهند که نیازهای مربوطه آنها را به طور دستی انتخاب کنند، اطلاعات مربوط به سنسور را به سرعت به آنها ارسال کنند. با وجودی که IoT توجه قابل توجهی را از سوی دانشگاه و صنعت به خود جلب کرده است، جستجو و انتخاب حسگر به طور جامع مورد توجه قرار نگرفته است. تکنیک های جستجو و انتخاب حسگر با استفاده از اطلاعات زمینه [6] به طور خاص مورد بررسی قرار نگرفته اند. یک نظرسنجی در مورد محاسبات آگاهانه متنی برای اینترنت چیزها [6] حسگر جستجو و انتخاب را به عنوان یک کار مهم در پیکره بندی حسگر خودکار و فرایندهای کشف زمینه شناسایی کرده است. یکی دیگر از بررسی های معنانشناسی برای اینترنت اشیاء [7] نیز یک منبع (به عنوان مثال، یک سنسور یا یک محرک) جستجو را که دارای قابلیت کشف است به عنوان یکی از مهمترین ویژگی هایی که در اینترنت اشیا مورد نیاز است، شناسایی کرده است. برنگی و همکاران [7] نیاز به حاشیه نویسی معنایی منابع و خدمات IoT را برجسته کرده اند. پردازش و تجزیه و تحلیل داده هایی که حاوی مفهوم معنایی هستند، عناصر ضروری برای پشتیبانی از جستجو و کشف [7] می باشند. این روش ما، رویکرد حاشیه نویسی سنسورها با اطلاعات زمینه مربوطه و استفاده از آن برای جستجو حسگر را تصدیق می کند. مثال های زیر نشان می دهد که راه حل های میان افزار IoT موجود، قابلیت جستجوی حسگر را فراهم می کند.

میان افزار متصل سنسور (LSM) [8] ، [9] انتخاب برخی از سنسورها و قابلیت جستجو را فراهم می کند. با این حال، آنها قابلیت های بسیار محدودی ، مانند انتخاب سنسورها بر اساس مکان و نوع حسگر دارند. همه جستجوها باید با استفاده از SPARQL انجام شود، که برای کاربران غیر فنی، کاربر پسند نیست. همانند LSM، چندین پروژه دیگر مرتبط با میان افزار IOT وجود دارد که در حال حاضر در حال توسعه است. GSN [3] پلت فرم یا پایه ایست که هدف آن ارائه میان افزار انعطاف پذیری برای رسیدگی به چالش های مربوط به ادغام داده ها و پردازش پرسوجوی

توزیع شده است. این موتور پردازش جریان داده عمومی است. GSN فراتر از تلاش های تحقیقاتی سنتی حسگر شبکه مانند مسیریابی، جمع آوری داده ها و بهینه سازی انرژی است. GSN لیست تمام سنسورهای موجود در جعبه دسته کوچکی که کاربران باید انتخاب کنند را فهرست می کند. با این حال، GSN به معنانشناسی برای مدل سازی ابرداده ها نیاز ندارد. یکی دیگر از رویکردها میکروسافت نقشه حسگر یا SensorMap [10] است. این تنها به کاربران اجازه می دهد سنسورها را با استفاده از نقشه محل، نوع حسگر و کلمات کلیدی انتخاب کنند. Xively (شرکت) یک رویکرد دیگر است که یک پلت فرم امن و قابل انطباق را فراهم می کند که دستگاه ها و محصولات را با برنامه ها متصل می کند تا کنترل زمان واقعی و ذخیره سازی داده ها را فراهم کند. این رویکرد همچنین فقط جستجوی کلیدواژه را فراهم می کند. تصاویری از عملکردهای جستجو ارائه شده توسط راه حل های IoT فوق در [11] ارائه شده است. راهکار پیشنهادی CASSARAM ما می تواند برای غنی سازی تمام راه حل های میان افزار IoT فوق با یک سنسور جستجوی جامع و دارای قابلیت انتخاب مورد استفاده قرار گیرد.

ما در ادامه، برخی از کارهای انجام شده در جستجو و انتخاب سنسور را توصیف می کنیم. ترانگ و همکاران [12] فازی مبتنی بر تکنیک جستجوی حسگر در مقایسه با نمره تقریبی جهت مقایسه خروجی یک سنسور داده شده با خروجی های چندین سنسور برای یافتن حسگر منطبق پیشنهاد می دهند. میر و همکاران [13] محل کارهای هوشمند / سنسورها را به عنوان زمینه اصلی محسوب می کنند و آنها را در یک ساختار منطقی شکل می دهند. سپس سنسورها با استفاده از تکنیک های جستجوی درختی به جستجوی مکان می پردازند. جستجوهای تحقیقی در مسیر یا مسیر گره درختی بصورت توزیعی پردازش می شود. الهی و همکاران [14] یک رویکرد جستجوگر حسگر مبتنی بر محتوا را پیشنهاد می کند (به عنوان مثال یافتن سنسوری که یک مقدار معین را در زمان پرس و جو نشان می دهد). Dyser موتور جستجوی پیشنهاد شده توسط Ostermaier و همکاران است. [15]، که برای اینترنت بی درنگ از چیزهایی که از مدل های آماری برای پیش بینی وضعیت اشیاء ثبت شده (حسگرها) استفاده می کند. هنگامی که کاربری یک پرس و جو را ارائه می کند، Dyser آخرین اطلاعات را برای شناسایی وضعیت فعلی واقعی به منظور تعیین اینکه آیا آن با پرس و جو کاربر مطابقت دارد، می کشد. مدل پیش بینی کمک می کند تا سنسورهای تطبیق با حداقل تعداد

بازیابی اطلاعات سنسور پیدا شوند. تعداد کمی از تلاش های مرتبط با آن بر اساس اطلاعات متنی در جستجوی اطلاعات حساس متمرکز شده است. پررا و همکاران [11] شباهت ها و تفاوت های موجود در بین جستجوی حسگر و جستجوی وب را مقایسه کرده اند. مشخص شد که اطلاعات زمینه نقش مهمی را در جستجو در خدمات وب (به ویژه در رابطه با خدمات وب) ایفا می کنند. بر طبق یک مطالعه در اروپا [16]، بیش از 12000 کار و خدمات مفید وب در مورد وب وجود دارد. حتی در چنین شرایطی، انتخاب بین جایگزین ها (بسته به ویژگی های زمینه) به یک مشکل چالش برانگیز تبدیل شده است. شباهت ها این استدلال را تقویت می کنند که انتخاب حسگر یک چالش مهم در همان سطح پیچیدگی به عنوان خدمات وب است. از سوی دیگر، تفاوت ها نشان می دهد که انتخاب سنسور به دلیل مقیاس IoT، یک چالش پیچیده تر در طول دهه آینده خواهد بود.

د و همکاران [17] یک معماری مفهومی، و پلت فرم IoT را برای حمایت از اشیاء دنیای واقعی و دیجیتال پیشنهاد کرده اند. آنها چندین مدل مبتنی بر هستی شناختی معنایی را ارائه داده اند که اجازه می دهد اطلاعات مربوط به منابع IoT (مثلا حسگرها، خدمات، عملگرها) را به دست آورند. با این حال، آنها بر سنسور متمرکز نیستند و تنها اطلاعات متنی که در نظر گرفته شده، محلی است. در مقابل، CASSARAM دقیقاً روی سنسورها تمرکز می کند و مجموعه جامعی از اطلاعات زمینه را در نظر می گیرد (نگاه کنید به بخش چهارم-F). گینارد و همکاران [18] کشف خدمات وب، پرس وجو، انتخاب و روش رتبه بندی را با استفاده از اطلاعات زمینه مربوط به دامنه IoT پیشنهاد کرده اند. به همین ترتیب، TRENDY [18] یک پروتکل کشف سرویس مبتنی بر رجیستری است که بر اساس سرویس وب مبتنی بر CoAP (پروتکل برنامه محدود شده) [20] با آگاهی از زمینه است. این پروتکل برای استفاده در دامنه وب چیزها (WoT) با هدف مقابله با تعداد زیادی از سرویس های وب (مثلا حسگرهای پیچیده شده در خدمات وب) پیشنهاد شده است. اطلاعات زمینه ای مانند تعداد ضربه، باتری و زمان پاسخ برای انتخاب خدمات استفاده می شوند. یک پیشنهاد جالب توسط Calbimonte و همکاران ارائه شده است [21] که یک رویکرد مبتنی بر هستی شناسی جهت دسترسی به داده ها و قابلیت جستجو برای جریان داده منابع را پیشنهاد کرده اند. این کار به کاربران اجازه می دهد تا نیازهای خود را در یک سطح مفهومی، مستقل از پیاده سازی بیان کنند. رویکرد CASSARAM

ما، می‌تواند برای تکمیل کار آنهایی استفاده شود که در آنجا ما از زمینه مبتنی بر جستجوی حسگر آنها پشتیبانی می‌کنیم و آنها دسترسی به داده‌های سنسور غنی شده معنایی را فراهم می‌کنند. علاوه بر این، نتایج ارزیابی ما می‌تواند برای درک مقیاس پذیری و عملکرد محاسباتی پارادایم داده‌های کار بزرگ خود به عنوان هر دو روش استفاده از هستی‌شناسی SSN مورد استفاده قرار گیرد. گارسیا کاسترو و همکاران [22] یک مدل هستی‌شناسی هسته‌ای برای زیرساخت‌های وب حسگر معنایی تعیین کرده‌اند. این می‌تواند برای مدل‌سازی شبکه‌های حسگر (با گسترش هستی‌شناسی (SSN)، منابع داده‌های سنسور و خدمات وب که منابع داده را نمایش می‌دهند، مورد استفاده قرار گیرد. رویکرد ما همچنین می‌تواند به صورت رویکرد uBox [23] به منظور جستجوی اشیاء در دامنه WoT با استفاده از اطلاعات متنی ادغام شود. در حال حاضر، uBox جستجوها را براساس برچسبهای موقعیت مکانی و شی (سنسور) دسته‌ها (انواع) انجام می‌دهند (مانند سلسله مراتب، دسته، محرک، سبک).

در جدول 1، تلاش‌های تحقیقاتی مختلفی که به چالش جستجوگر حسگر می‌پردازد خلاصه می‌شود. جدول یک لیست تلاش‌ها و تعداد سنسورهای مورد استفاده در آزمایش‌های آنها را نشان می‌دهد.

3. تعریف مسئله و انگیزه

مسائلی که ما در این مقاله به آن اشاره می‌کنیم، می‌تواند به شرح زیر تعریف شود. با توجه به افزایش تعداد سنسورهای موجود، ما باید سنسورهایی را انتخاب کنیم که داده‌هایی را ارائه می‌دهند که با بهترین و کارآمدترین روش به حل مسئله از نزدیک کمک می‌کنند. هدف ما این نیست که مسائل کاربران را حل کنیم، بلکه هدف برای کمک به آنها در جمع‌آوری داده‌های سنسور می‌باشد. کاربران می‌توانند این داده‌ها را به طرق مختلف خود برای حل مسائل بیشتر پردازش کنند. ما به منظور دستیابی به این هدف، باید سنسورها را براساس بخش‌های مختلف اطلاعات متنی جستجو و انتخاب کنیم. ما دو دسته از الزامات را به طور عمده مشخص می‌کنیم: الزامات مبتنی بر نقطه (غیر قابل بحث) و الزامات مبتنی بر مجاورت (قابل بحث). ما با ارائه صحنه‌ها و چالش‌های کاربردی دنیای واقعی، مشکل را با جزئیات در [11] مورد بررسی قرار دادیم.

جدول 1: تعداد سنسورهای مورد استفاده در ارزیابی‌های تجربی با استفاده از روش‌های مختلف جستجوی حسگر

رویکرد	تعداد حسگرهای استفاده شده در آزمایشات
Truong et al. [13]	42
Elahi et al. [14]	250
Ostermaier et al. [15]	385
Mayer et al. [13]	600
Calbimonte et al. [24] ²	1400
LSM [9]	100,000

اول اینکه، الزامات مبتنی بر نقطه که باید به طور کامل اجرا شود، وجود دارد. به عنوان مثال، اگر کاربر به اندازه گیری دما در یک مکان خاص علاقه مند باشد (به عنوان مثال، کانبرا)، نتیجه (به عنوان مثال لیست سنسورها) باید حاوی سنسورهایی باشد که بتوانند دما را اندازه گیری کنند. کاربر با ارائه هر گونه نوع دیگر سنسور (مانند سنسورهای فشار) نمی تواند از کاربر استفاده کند. در این نوع شرایط الزامی هیچ معامله یا سازشی وجود ندارد. محل را می توان به عنوان یک نیاز مبتنی بر نقطه شناسایی کرد. دومین الزامات مبتنی بر نزدیکی است که باید در بهترین حالت ممکن اجرا شود. با این حال، ملاقات با نیاز کاربر دقیق لازم نیست. کاربران ممکن است تمایل داشته باشند با یک تفاوت کوچک (تغییری) روبرو شوند. به عنوان مثال، کاربر علاقه مند به همان شرایط قبل است. با این حال، در این وضعیت، کاربر علاوه بر الزامات مبتنی بر نقطه، نیازهای مبتنی بر نزدیکی را نیز فراهم می کند. کاربر ممکن است سنسورهایی با دقت حدود 92٪ و اطمینان 85٪ را درخواست کند. بنابراین، کاربر به این ویژگی ها بالاترین اولویت را می دهد. کاربر سنسورهایی را می پذیرد که تقریباً این نیازها را برآورده کند، حتی اگر همه ویژگی های دیگر مطلوب نباشد (به عنوان مثال هزینه خرید ممکن است بالا باشد و پاسخ حسگر ممکن است کند باشد). توجه به این نکته که ممکن است کاربران قادر به ارائه هیچ خاصیت خاصی نباشند نیز حائز اهمیت است، بنابراین سیستم باید بتواند اولویت های کاربر را درک کند و نتایج را با استفاده از مقایسه فراهم کند.

یکی دیگر از انگیزه‌های تحقیقی ما آمار و پیش بینی هایی است که رشد سریع استقرار سنسور مربوط به IoT و شهرهای هوشمند را نشان می دهد. تخمین زده می شود که امروزه حدود 1.5 میلیارد کامپیوتر با اینترنت فعال و بیش از 1 میلیارد تلفن همراه فعال شده‌اند. تا سال 2020، 50 تا 100 میلیارد دستگاه متصل به اینترنت وجود خواهد

داشت [25]. علاوه بر این، کار ما با روند افزایشی توسعه راه حل های میان افزار IoT تحریک کننده است. امروزه، بسیاری از راه حل های پیشرو میان افزار تنها به جستجو حسگر و انتخاب قابلیت حسگر، همانطور که در بخش دوم ذکر شده است، محدود شده است.

ما اهمیت قابلیت جستجوی حسگر را با استفاده از برنامه های جاری و بالقوه برجسته می کنیم. پروژه های هوشمند کشاورزی [26] مانند Phenonet [27] اطلاعات را از هزاران سنسور جمع آوری می کند. همانطور که در بخش چهارم-F ذکر شده است، هر سنسور ممکن است با توجه به ناهمگونی یا عدم تجانس، دارای مقادیر مختلف زمینه باشد. اطلاعات متنی را می توان برای انتخاب درست سنسورها بسته به الزامات و شرایط مورد استفاده قرار داد. به عنوان مثال، CASSARAM فقط وقتی به بازیابی داده ها از سنسورها کمک می کند که سنسورهای جایگزین در دسترس انرژی بیشتری به جا گذاشته باشند. چنین اقداماتی کمک می کند تا کل شبکه حسگر را برای مدت زمان بسیار طولانی تر بدون بازپس گیری و شارژ کردن به کار گیرند. سنجش به عنوان یک مدل سرویس معماری [28] یک دوره زمانی را فراهم می کند که اطلاعات حسگر را از طریق ابر منتشر و معامله می کند. مصرف کنندگان (به عنوان مثال، کاربران) مجاز به انتخاب تعدادی از سنسورها هستند و در مدت زمانی که به توافق می رسند داده ها را با پرداخت مبلغی بازیابی می کنند. در چنین شرایطی، اجازه دادن به مصرف کنندگان برای انتخاب سنسورهایی که می خواهند بر اساس اطلاعات متنی باشد، بسیار مهم است. به عنوان مثال برخی از مصرف کنندگان، ممکن است تمایل به پرداخت بیشتر برای اطلاعات بسیار دقیق (به عنوان مثال سنسور بسیار دقیق) داشته باشند، در حالی که دیگران ممکن است بسته به نیازها، شرایط و تنظیمات آنها مایل به پرداخت کمتر برای داده های دقیق تر باشند.

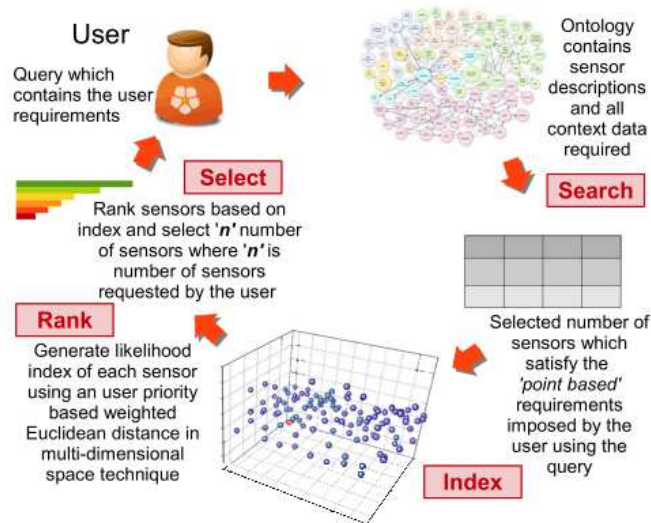
4. جستجوی حسگر متنی آگاهانه، مدل رتبه بندی و انتخاب

در این بخش، روش انتخاب سنسور پیشنهاد شده به روش گام به گام را به طور دقیق ارائه می کنیم. نخست، یک نمای کلی سطح بالا از مدل ارائه می دهیم، که شرح جریان کلی و عمق بحرانی را نشان می دهد. سپس، توضیح می دهیم که چگونه ترجیحات کاربر ضبط می شود. بعد، مدل ارائه نمای داده و توسعه پیشنهاد شده ارائه می شود. در نهایت، تکنیک های پرس و جو معنایی و استدلال کمی با کمک برخی از الگوریتم ها مورد بحث قرار می گیرد. تمام الگوریتم های

ارائه شده در این مقاله خود توضیحی هستند و الگوریتم های رایج مورد استفاده در این مقاله در جدول 2 ارائه شده است.

جدول 2: جدول علامت گذاری الگوریتمی عمومی

نماد	تعریف
O	هستی شناسی شامل توصیفات حسگرها و مقدار محتویات متنی مربوط به همه حسگرها می باشد.
P	مجموعه اولویت کاربر شامل مقدار اولویت در تمام ویژگی های متنی است.
Q	جستجو شامل الزامات مبتنی بر نقطه بیان شده در SPARQL است.
N/N_{All}	تعداد حسگرهای مورد نیاز کاربر یا تعداد کلی حسگرها قابل دسترس است.
$S_{Filtered}$	شامل نتایج جستجوی Q است.
$S_{Results}$	تنظیم نتایج شامل تعداد حسگرهای انتخاب شده است.
$S_{Indexed}$	مجموعه شاخص سنسور مقادیر شاخص سنسور را ذخیره می کند.
M	فضای چند بعدی که در آن هر ویژگی متن با یک بعد و حسگرهای ترسیم شده نمایش داده می شود.
UI	ورودی کاربر که شامل مقادیر ورودی ارائه شده توسط کاربران از طریق رابط کاربر است.
SC/SC	مقدار تمام لغزنده ها و مقدار یک لغزنده
P^w	شامل مقدار اولویت کاربر است که به طور نرمال بارگیری شده است.
P_i/P_i^w	مقدار ویژگی متنی i^{th} یا مقدار ویژگی متن i^{th} به شکل نرمال.
CP/CP	مجموعه اولویت متنی شامل تمام اطلاعات متنی یا مقدار ویژگی متنی i^{th} می باشد.
NCP	مجموعه ویژگی متن نرمال
M	خطای جانبی
S_j	این حسگر j^{th} است.
$CP_i S_j$	مقدار CP از ویژگی i^{th} حسگر j^{th}
CP^{ideal}	مقدار CP حسگرهای ایده آلی که کاربر ترجیح می دهد.



شکل 1: بررسی کلی سطح بالای CASSARAM

User: کاربر: پرس و جویی که شامل شرایط مورد نیاز کاربر می باشد.

Select: انتخاب: درجه بندی حسگرها بر اساس شاخص و انتخاب تعداد "n" حسگرها در جایی که "n" تعداد حسگرهای درخواست شده توسط کاربر است.

Rank: درجه بندی: ایجاد شاخص های احتمالی هر حسگر با استفاده از اولویت کاربر بر اساس فاصله اقلیدسی وزن در تکنیک فضای چند بعدی.

Search: تحقیق: هستی شناسی شامل توصیفات حسگر و تمام داده های متنی مورد نیاز است.

Index: شاخص: تعداد حسگرهای منتخبی که الزامات مبتنی بر نقطه تحمیلی را با استفاده از پرس و جوی کاربر برآورده می کند.

A. بررسی اجمالی مدل سطح بالا

مراحل بحرانی CASSARAM در شکل 1 ارائه شده است. همانطور که قبلا ذکر شد، هدف ما این است که کاربران قادر به جستجو و انتخاب سنسورهایی باشند که به بهترین وجه نیاز آنها را برآورده می کند. در مدل ما الزامات کاربر از نظر کاربر به دو دسته تقسیم می شود: الزامات مبتنی بر نقطه و الزامات مبتنی بر نزدیکی، همانطور که در بخش سوم بحث شده است. الگوریتم 1 جریان اجرای CASSARAM را توصیف می کند. در ابتدا، CASSARAM

نیازهای مبتنی بر نقطه، نیازهای مبتنی بر نزدیکی و اولویت های کاربر را شناسایی می کنند. اولاً، کاربران باید نیازهای مبتنی بر نقطه را انتخاب کنند. به عنوان مثال، یک کاربر ممکن است بخواهد اطلاعات سنسور را از 1000 سنسور دما مستقر در کانبرا جمع آوری کند. در این وضعیت، نوع سنسور (به عنوان مثال، درجه حرارت)، موقعیت مکانی (به عنوان مثال، کانبرا) و تعداد سنسورهای مورد نیاز (یعنی 1000) الزامات مبتنی بر نقطه هستند. ابزار نمونه اولیه CASSARAM ما یک رابط کاربری را برای بیان این اطلاعات از طریق پرس و جویهای SPARQL فراهم می کند. در CASSARAM، هر زمینه مالکیت می تواند به یک نیاز اساسی تبدیل شود. بعد، کاربران می توانند نیازهای مبتنی بر نزدیکی را مشخص کنند. تمام خواص زمینه‌ای که در بخش چهارم-F ارائه می شود، در مقیاس تطبیقی با تنظیم اولویت‌ها از طریق یک رابط کاربری لغزنده، در شکل 2 نشان داده شده است. بعد، هر سنسور در یک فضای چند بعدی طراحی شده است که در آن هر بعد ابعاد متنی را مشخص می کند (مانند دقت، قابلیت اطمینان، زمان تاخیر یا نهفتگی). همانطور که در الگوریتم 3 توضیح داده شده است، هر بعد نرمالیزه یا بهنجار شده است [0,1]. سپس شاخص وزنی مبتنی بر تطبیق-اولویت (CPWI) برای هر سنسور بوسیله ترکیب اولویت های کاربر و مقادیر محتویات ایجاد می شود که در بخش چهارم-E توضیح داده شده است. سنسورها با استفاده از CPWI رتبه بندی می شوند و تعداد سنسورهای مورد نیاز کاربر از بالای لیست انتخاب شده است.

B. اولویت‌های کاربر

همانطور که در شکل 2 نشان داده شده است، این تکنیکی است که ما با استفاده از یک رابط کاربری برای اولویت کاربر ترسیم کردیم. CASSARAM به کاربران امکان می دهد که ویژگی متنی را که در مقایسه با دیگران برای آنها مهم تر است را بیان کنند. اگر یک کاربر نمی خواهد یک خاصیت متنی خاص را در فرآیند نمایه سازی در نظر بگیرد، می تواند با انتخاب جعبه چک مرتبط با این ویژگی خاص متنی از آن اجتناب کند. به عنوان مثال، طبق شکل 2، هنگام محاسبه CPWI انرژی در نظر گرفته نخواهد شد. این به این معنی است که کاربر مایل به پذیرش سنسورها با هر سطح مصرف انرژی است. در صورتی که ویژگی متنی برای کاربران اهمیت دارد، کاربران باید نوار لغزنده هر ویژگی زمینه را قرار دهند. مقیاس لغزنده از 1 شروع میشود، به این معنی که هیچ اولویتی (یعنی گوشه سمت چپ) ندارد.

بالاترین اولویت را می توان توسط کاربر به صورت ضروری با کمک یک مقیاس گذار تنظیم کرد، جایی که یک مقیاس بالاتر لغزنده ها را حساس تر می کند (e.g. $10^2 = 1$ الی 10^3 ، 10^4 ، 100). الگوریتم 2 فرایند ضبط کاربر اولویت را توصیف می کند.

همانطور که در شکل 2 نشان داده شده است، اگر کاربر می خواهد قابلیت اطمینان یک سنسور نسبت به دقت آن بیشتر احساس شود یا وزن بیشتری داشته باشد، موضوع قابلیت اطمینان باید بیشتر از موضوع لغزندگی مورد توجه قرار گیرد. وزن برای هر ویژگی زمینه محاسبه می شود. بنابراین، اولویت بالاتر به معنی وزن بیشتر است. در نتیجه، سنسورهایی با قابلیت اطمینان و دقت بالا به دقت رتبه بندی می شوند. با این حال، ممکن است این سنسورها با توجه به اولویت کم در مورد هزینه، هزینه های بالایی داشته باشند.

الگوریتم 1 جریان اجرای CASSARAM
نیاز: (M), (N), (Q), (P), (O)
1: خروجی: S Results
2: S Filtered → هستی شناسی پرس و جو (O,Q)
3: اگر اصل اینست (S filtered) > N پس
4: بازگشت: S Filtered → S Results
5: دیگر اینگه
6: p → گرفتن اولویت های کاربر
7: M → طراحی حسگرها در فضای چند بعدی
8: S index → محاسبه CPWI (S Results , M)
9: S results → درجه بندی حسگرها

10: Results $S \rightarrow$ انتخاب حسگرها
11: بازگشت Results S
12: و اگر

الگوریتم 2 اولویت کاربر
نیاز: (UI)، (SC)
1: خروجی: p^w
2: $P \rightarrow$ استخراج اولویت های کاربر
3: بالاترین SC \rightarrow دریافت حداکثر اولویت (SC)
4: پایین ترین SC \rightarrow دریافت حداقل اولویت (SC)
5: درجه بندی SC \rightarrow پایین ترین SC - بالاترین SC
6: برای هر اولویت ویژگی زمینه $P \in P_i$ انجام می شود
7: $(p_i \div S_{range}) \rightarrow p_i^w$
8: اگر $0 \leq p_i^w$ پس
9: p_i^w به p^w اضافه می شود
10: دیگر اینکه
11: اگر ادامه یابد
12: و اگر
13: پایان دادن
14: بازگشت p^w



شکل 2. وزن W_1 به ویژگی قابلیت اطمینان اختصاص دارد. وزن W_2 به ویژگی دقت اختصاص داده شده است. وزن W_3 به ویژگی در دسترس پذیری اختصاص داده شده است. وزن W_4 ، وزن پیش فرض، به ویژگی هزینه اختصاص دارد. اولویت بالا به این معنی است که همیشه مطلوب است و اولویت کم همیشه نامطلوب است. به عنوان مثال، اگر کاربر هزینه ای برای اولویت بالا پرداخت کند (بیشتر به راست)، یعنی CASSARAM سعی دارد حسگرهایی را که داده ها را با کمترین هزینه تولید می کنند، پیدا کند. به همین ترتیب، اگر کاربر دقت اولویت بالا را تعیین کند، به این معنی است که CASSARAM سعی می کند تا حسگر هایی که داده ها را با دقت بالا تولید می کنند، پیدا کند.

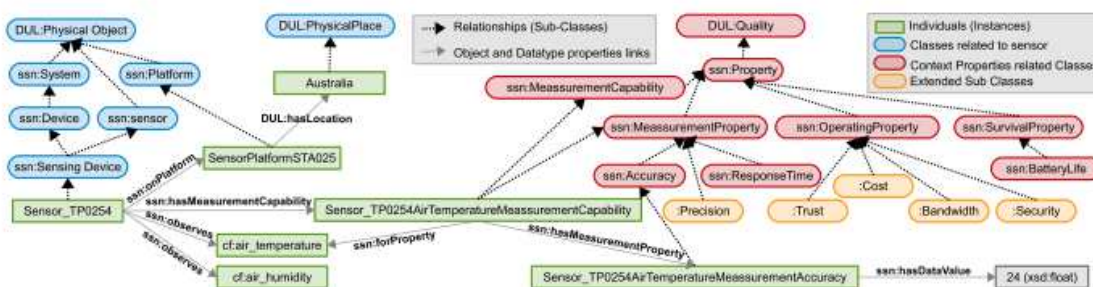
C. مدل سازی داده ها و نمایندگی

در این مقاله، ما هستی شناسی شبکه سنسور مفهومی (SSN) [5] را برای مدل سازی توصیف های حسگر و ویژگی های زمینه (یا متنی) مورد استفاده قرار دادیم. دلایل اصلی انتخاب هستی شناسی SSN، قابلیت همکاری آن و روند استفاده از هستی شناسی در دامنه مدیریت داده های IoT و سنسور یا حسگر است. مقایسه هستی شناسی حسگرهای معنایی مختلف در [29] ارائه شده است. هستی شناسی SSN قادر به مدل سازی مقدار قابل توجهی از اطلاعات در مورد سنسورها، مانند قابلیت سنسور، عملکرد، شرایطی است که می تواند مورد استفاده قرار گیرد، و غیره. جزئیات در [5] ارائه شده است. هستی شناسی SSN شامل شایع ترین خواص زمینه مانند صحت، دقت، راندگی، حساسیت، انتخابی، محدوده اندازه گیری، محدودیت تشخیص، زمان پاسخ، فرکانس و تاخیر است. با این حال، می توان هستی شناسی SSN را با طبقه بندی نامحدود به سه طبقه گسترش داد: ویژگی اندازه گیری، ویژگی عملیاتی، و ویژگی بقاء. ما بخش ساده ای از هستی شناسی SSN در شکل 3 را نشان می دهیم. همانطور

که در بخش چهارم-F شده است، ما با اضافه کردن چند زیر مجموعه بر اساس چارچوب زمینه، کیفیت طبقات را بالا می بریم. تمام مقادیر محتویات زمینه در هسته شناسی SSN و در واحدهای اندازه گیری اصلی آنها ذخیره می شود. CASSARAM به جهت حصول اطمینان آنها را بر اساس تقاضا [0,1] عادی می کند و به حالت نرمال در می آورد. تکنیک های ذخیره سازی می توانند برای افزایش عملکرد اجرایی استفاده شوند. با توجه به پیشرفت های تکنولوژیکی در توسعه سخت افزار سنسور، محدودیت های بالایی و پایینی برای برخی از ویژگی های زمینه غیرممکن است (مثلا عمر باتری در طول زمان با پیشرفت فن آوری های سخت افزاری سنسور بهبود می یابد). بنابراین، ما الگوریتم 3 را پیشنهاد می کنیم تا به طور پویا خواص زمینه را عادی سازد و به حالت نرمال درآورد.

D. فیلتر کردن با استفاده از دلیل یابی پرس و جو

هنگامی که نیازهای کاربر مبتنی بر نقطه مشخص شده اند، باید با استفاده از SPARQL بیان شوند. پرس و جوی معنایی دارای نقاط ضعف و نقاط قوت است. هنگامی که پرس و جو پیچیده می شود، عملکرد کاهش می یابد [30]. فیلترینگ مبتنی بر بیان رابطه ای نیز می تواند مورد استفاده قرار گیرد؛ با این حال، استفاده از آن الزامات محاسباتی را افزایش می دهد. توضیحات بیشتر در بخش چهارم-B ارائه شده است. هر یک از خصوصیات زمینه که در قسمت پنجم-F مشخص شده است، می تواند الزامات مبتنی بر نقطه باشد و باید در SPARQL نمایش داده شود. این مرحله S Filtered را تولید می کند، در جایی که تمام سنسورها تمام نیازهای مبتنی بر نقطه را برآورده می کنند.



شکل 3. مدل داده مورد استفاده در CASSARAM. در هستی شناسی SSN، سنسورها به دستگاههای سنجش فیزیکی محدود نمی شوند؛ در واقع یک سنسور چیزی است که می تواند ارزش یک پدیده را برآورد یا محاسبه کند، بنابراین یک دستگاه پردازش یا ترکیبی محاسباتی می تواند نقش یک حسگر را بازی کند. یک دستگاه حسگر وسیله

ایست که سنجش را تکمیل می کند [5]. دستگاه حسگر نیز زیر مجموعه یک سنسور است. با توجه به تعریف فوق، تمرکز ما بر حسگرها است. هسته شناسی CF (آب و هوا و پیش بینی) حوزه خاصی از هستی شناسی خارجی را دربرمی گیرد. هستی شناسی DOLCE+DnS Ultralite (DUL) مجموعه ای از مفاهیم سطح بالایی را فراهم می کند که می تواند پایه ای برای قابلیت همکاری در بین بسیاری از هستی شناختی های سطح متوسط و پایین باشد. جزئیات بیشتر در [5] ارائه شده است.

الگوریتم 3 نرمال سازی پویای flexi
نیاز: (CP), (S), (cp i)
1: خروجی : NCP
2: $cp_i^{Sj} \rightarrow$ دریافت مقادیر اولویت جدید
3: بالاترین $cp_i \rightarrow$ بالاترین بازیابی (CP)
4: پایین ترین $cp_i \rightarrow$ بالاترین بازیابی (CP)
5: اگر بالاترین $cp_i > cp_i^{Sj}$ پس
6: بالاترین $cp_i \rightarrow cp_i^{Sj}$
7: برای هر $cp_i^{Sj} \in CP, S$ انجام می شود
8: به روز رسانی (NCP) $\rightarrow \left[\frac{(cp_i^{Sj} - cp_i^{پایینترین})}{cp_i^{بالاترین} - cp_i^{پایینترین}} \right]$
9: در آخر
10: دیگری
11: به روز رسانی (NCP) $\rightarrow \left[\frac{(cp_i^{Sj} - cp_i^{پایینترین})}{cp_i^{بالاترین} - cp_i^{پایینترین}} \right]$
12: و اگر

* حسگرهای ثبت شده در میان افزار IOT

E. رتبه بندی با استفاده از استدلال کمی

در این مرحله سنسورها بر اساس نزدیکی الزامات کاربر رتبه بندی می شوند. ما یک روش ارزیابی مبتنی بر فاصله وزنی اقلیدوسی به نام شاخص تورم مبتنی بر اولویت مقایسه (CPWI) ایجاد کرده ایم که به شرح زیر است:

$$(CPWI) = \sqrt{\sum_{i=1}^n [w_i (U_i^d - S_i^a)^2]}$$

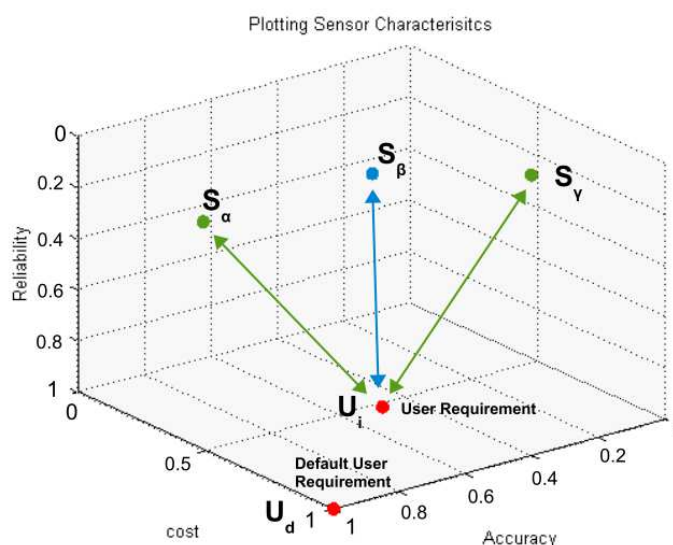
اول اینکه، هر سنسور در فضای چند بعدی طراحی شده است که در آن هر موضوع زمینه با یک بعد نشان داده شده است. سپس، کاربران می توانند با وارد کردن دستی مقادیر محتویات زمینه که در شکل 4 توسط U_i نشان داده شده است، یک حسگر ایده آل را در فضای چند بعدی نشان دهند. CASSARAM پیش فرض، به طور خودکار حسگر ایده آل را همانطور که در U_d نشان داده می شود (یعنی بالاترین مقدار برای همه زمینه ها خواص) ترسیم می کند. بعد، اولویت های تعیین شده توسط کاربر بازیابی می شوند. بر اساس موقعیت لغزنده ها (در شکل 2)، وزن ها به صورت مقایسه ای محاسبه می شوند. الگوریتم 4 روند شاخص گذاری را توصیف می کند. این CPWI را محاسبه می کند و سنسورها را با استفاده از تکنیک های معکوس عادی که به ترتیب نزولی هستند رتبه بندی می کند. CASSARAM سنسور N را از بالا انتخاب می کند.

شکل 4. سنسورها به منظور نمایش در فضای سه بعدی طراحی شده اند. S_α, S_β و S_γ نشانگر سنسورهای

واقعی هستند. U_i نشان دهنده سنسور مورد نظر کاربر است. U_d نشان دهنده حسگر پیش فرض کاربر است.

CPWI محاسبه فاصله وزنی بین $S_j = \alpha // \beta // \gamma$ و $U_i // d$ است. و کوتاه ترین فاصله یعنی سنسوری که

چون نزدیک به نیاز کاربر است رتبه بالاتری خواهد داشت.



F. چارچوب مفهومی

پس از ارزیابی تعدادی از تلاش های تحقیقاتی انجام شده در زمینه کیفیت خدمات مربوط به سرویس های وب [31]، محاسبات موبایل [32]، جمع آوری داده های تلفن همراه [33] و هستی شناسی های حسگر [5]، و ویژگی های زیر که شامل ذخیره و نگهداری در ارتباط با هر سنسور است را استخراج کردیم. این اطلاعات کمک می کند تا تصمیم بگیرند که کدام حسگر در یک وضعیت خاص مورد استفاده قرار گیرد. ما تعریف زیر را برای این مقاله می پذیریم. "زمینه هر گونه اطلاعاتی است که می تواند برای توصیف وضعیت یک موجودیت یا هویت استفاده شود. موجودیت یک شخصیت یک فرد، مکان یا شیء است که در رابطه با تعامل بین یک کاربر و یک برنامه کاربردی، از جمله کاربر و برنامه های کاربردی خود، مرتبط است." [34]. CASSARAM هیچ محدودیتی در تعداد خواص زمینه ای که می تواند مورد استفاده قرار گیرد، وجود ندارد. اطلاعات زمینه ای بیشتر در صورت لزوم می تواند به لیست زیر اضافه شود. چارچوب زمینه ما شامل دسترسی، درستی، قابلیت اطمینان، زمان پاسخ، فرکانس، حساسیت، محدوده اندازه گیری، قابلیت انتخاب، دقت، زمان تاخیر، راندگی، وضوح، حد تشخیص، محدوده توان عملیاتی، عمر سیستم (سنسور)، عمر باتری، امنیت، دسترسی، استقامت، مدیریت استثنائی، قابلیت همکاری، قابلیت اطمینان، امتیاز رضایت کاربر، ظرفیت عملکرد، هزینه انتقال داده، هزینه تولید اطلاعات، هزینه مالکیت داده، پهنای باند و اعتماد است.

الگوریتم 4 شاخص وزنی مبتنی بر اولویت تطبیقی

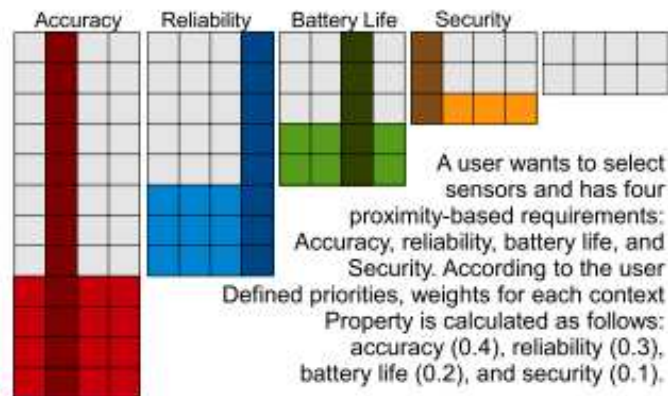
نیاز: $(UI), (P^{sj}), (S^{indexed}), (CP), (P^w)$
1: خروجی: S^{Ranked}
2: $CP^{Ideal} \rightarrow$ الزامات تقریبی (UI)
3: طراحی در فضای چند بعدی (CP^{ideal})
4: برای هر حسگر $S \in S_j$ انجام می شود
5: طراحی در فضای چند بعدی (CP^{Sj})
6: در آخر
7: فرمول شاخص گذاری (S^α) برای $S^\alpha = \sqrt{\sum_{i=1}^n [(U_i^d - S_i^\alpha)^2]}$
8: برای هر حسگر $S \in S_j$ انجام می شود
9: $S^{indexed} \rightarrow$ درجه بندی معکوس نرمال $(S^{indexed})$ * یعنی: پایین ترین مقداری که بالاتر درجه بندی می شود نشان دهنده فاصله موزون بین حسگر مورد نظر کاربر و حسگر واقعی است
10: بازگشت: S^{Ranked}

الگوریتم 5 فیلترینگ غیرمستدل مبتنی بر اولویت تطبیقی
نیاز: $(O), (P), (Q), (N), (M/)$
1: خروجی: $S^{Filtered}$
2: $S \rightarrow$ هستی شناسی پرس و جو (O, Q)
3: $P^w \rightarrow$ بدست آوردن اولویت وزنی (P)
4: $P^{percentages} \rightarrow$ تبدیل وزن به درصد (P^w)
5: $N_{ALL} \rightarrow$ تعداد کلی حسگرهای قابل دسترس (P^w)
6: $N \rightarrow$ تعداد حسگرهای مورد نیاز (UI)
7: $N_{Removable} \rightarrow (N_{All} - N)$
8: $P^{percentages} \rightarrow$ ترتیب نزولی $(P^{percentages})$
9: برای هر درصد اولویت $P \in P^{percentage}$ انجام می شود
10: $S^{Filtered} \rightarrow$ پرسش $S^{Filtered}$ و مرتب شده توسط P

11: حذف کردن $N_{Removable} \times (100-M)$ حسگرها از پایین

12: در آخر

13: بازگشت $S_{Filtered}$



شکل 5. شکل بصری فیلترینگ غیرمستدل مبتنی بر اولویت تطبیقی

- مصرف کننده می خواهد حسگر را انتخاب کند و چهار الزام تقریبی دارد : دقت، اعتبار، عمر باتری، و امنیت. با توجه به تعریف اولویت های کاربر وزن هر ویژگی زمینه به شرح زیر محاسبه می شود: دقت (0.4)، قابلیت اطمینان (0.3)، عمر باتری (0.2)، و امنیت (0.1).

5. بهبود مطلوبیت و کارآمدی

در این بخش، ما سه رویکرد را ارائه می دهیم که کارآمدی و توانایی CASSARAM را بهبود می بخشد. اول، ما یک رویکرد غیرمستدل را پیشنهاد می کنیم که می تواند تعداد زیادی از سنسورها را با دقت به فروش برساند. دوم، ما تکنیک فیلترینگ مبتنی بر بیان رابطه ای را پیشنهاد می کنیم که منابع محاسباتی را ذخیره می کند. سوم، ما نسبت به چالش جستجو و انتخاب حسگر توزیع شده رسیدگی می کنیم.

A. فیلترینگ غیر مستدل مبتنی بر اولویت-تطبیقی (CPHF)

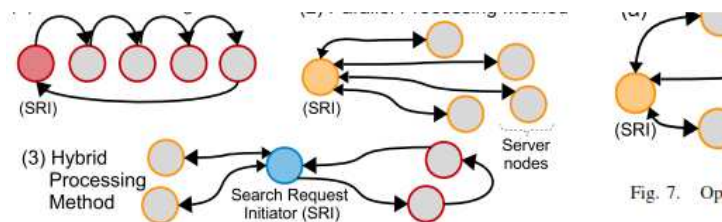
تا کنون راه حل مورد بحث ما در مورد تعداد کمی از سنسورها بوده است. با این حال، هنگامی که تعداد سنسورهای موجود برای جستجو افزایش می یابد، این مدل نامناسب می شود. اجازه دهید مثالی برای تشخیص ناکارآمدی در نظر

بگیریم. فرض کنید ما به یک میلیون سنسور دسترسی داشته باشیم. یک کاربر می خواهد 1000 سنسور را از آنها انتخاب کند. در چنین شرایطی، CASSARAM یک میلیون سنسور را با استفاده از الزامات مبتنی بر نزدیکی ارائه شده توسط کاربر رتبه بندی و شاخص گذاری کرده و 1000 سنسور را انتخاب می کند. با این حال، شاخص گذاری و رتبه بندی تمامی سنسورهای ممکن (در این مورد یک میلیون) ناکارآمد است و مقدار قابل توجهی از منابع محاسباتی را از بین می برد. علاوه بر این، CASSARAM به دلیل چنین مورد نامحدود قادر به پردازش تعداد زیادی از پرس و جوهای کاربر نخواهد بود. ما یک تکنیک به نام فیلترینگ اکتشافی (غیر مستدل) مبتنی بر اولویت-تطبیقی (CPHF) را پیشنهاد می کنیم تا CASSARAM را به طور موثرتری به کار بندیم. فرایند اجرا در الگوریتم 5 توضیح داده شده است. ایده اصلی این است که سنسورهایی را که دور از حسگر ایده آل تعیین شده توسط کاربر قرار گرفته اند را حذف کرده و تعداد سنسورهایی را که باید شاخص گذاری و رتبه بندی شوند را کاهش دهیم. شکل 5 رویکرد CPHF را با یک سناریوی نمونه نشان می دهد. رویکرد CPHF می تواند به شرح زیر توضیح داده شود. ابتدا تمام سنسورهای واجد شرایط به ترتیب نزولی بر حسب بالاترین خصوصیات بافت وزن (در این مورد دقت) رتبه بندی می شوند. سپس، 40٪ (از $N_{Removable}$ قابل جداشدن) سنسورها از پایین لیست حذف می شوند. بعد، سنسورهای باقیمانده باید به ترتیب نزولی بر حسب بالاترین خصوصیات بعدی بافت وزن (در این مورد اطمینان) مرتب شوند. سپس 30٪ (از $N_{Removable}$ قابل جداشدن) از سنسورهای پایین لیست باید حذف شوند. این فرآیند باید برای خواص باقیمانده نیز مورد استفاده قرار گیرد. در نهایت، سنسورهای باقیمانده باید شاخص گذاری و رتبه بندی شوند. این رویکرد به طور چشمگیری شاخص گذاری و رتبه بندی ناکارآمد مربوطه را کاهش می دهد. به طور گسترده، این دسته از تکنیک ها انتخاب Top-K نامیده می شوند که در هر تکرار، سنسورهای بالا انتخاب می شوند. اثربخشی این رویکرد در بخش هفت مورد بحث و بررسی قرار می گیرد.

B. فیلترینگ مبتنی بر بیان - رابطه ای (REF)

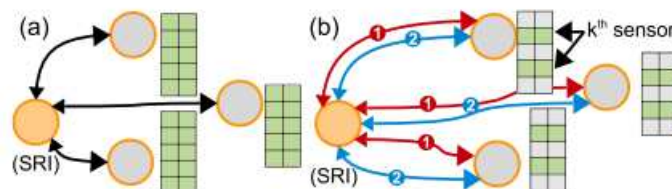
این بخش توضیح می دهد که چگونه منابع محاسباتی را می توان ذخیره کرد و چگونه با اجازه دادن به کاربران برای تعیین ارزش محتویات مورد نظر با استفاده از اپراتورهای ارتباطی مانند $<$ ، $>$ و \leq ، سرعت پردازش جستجو و

انتخاب حسگر را مشخص کرد. به عنوان مثال، کاربران با استفاده از اپراتورهای ارتباطی می توانند حد بالا، حد پایین یا هر دو را تعریف کنند. تمام خصوصیات متفاوتی که توسط اپراتورهای ارتباطی تعریف شده است، به غیر از علامت برابر (=)، به عنوان الزامات تقریباً غیر قابل انتقال در نظر گرفته می شوند. با توجه به CASSARAM، نیازمندی های غیر قابل انتقال و همچنین تقریباً غیر قابل انتقال با استفاده از پرسش های معنایی تعریف می شوند. اجازه دهید سناریویی را در نظر بگیریم که کاربر می خواهد سنسورهایی را انتخاب کند که دارای 85٪ دقت باشند. با این حال، کاربر می تواند با ارائه سنسور ها با دقت بین 70٪ و 90٪ راضی باشد. چنین الزاماتی مورد نیاز سمینارهای قابل مذاکره یا قابل انتقال هستند. تعریف چنین محدوده کمک می کند تا در مرحله پرس و جو معنایی سنسورهای نامناسب بدون حتی بازیابی آنها در مرحله تولید CPWI نادیده گرفته شود، و این موجب صرفه جویی منابع محاسباتی می شود. اگر چه کاربران ممکن است محدوده را تعیین کنند، اما سنسورها با توجه به اولویت کاربر با استفاده از مفاهیم و قوانین مشابه که در بخش چهارم توضیح داده می شوند، رتبه بندی می شوند. اثربخشی این روش در بخش هفت ارزیابی می شود.



شکل 6. روش های توزیع شده پردازش در CASSARAM

- روش پردازش زنجیره ای (2) روش پردازش موازی (3) روش پردازش مرکب



شکل 7. بیهنه سازی: (الف) بدون انبساط یا کشش K و (ب) با کشش K

C. جستجوی حسگر توزیع شده

ما توضیح دادیم که چگونه CASSARAM در یک محیط جداگانه بدون توجه به ماهیت توزیع شده مشکل کار می کند. در حالت ایده آل، ما انتظار داریم که تمام سنسورها به یک سرور مجزا متصل نشوند (به عنوان مثال، نمونه یک میان افزار تنها). بدین ترتیب، ذخیره توصیفات کامل حسگرها و اطلاعات متنی مربوط به آن در بسیاری از سرورهای مختلف به شیوه ای بیهوده بسیار ناکارآمد است. به طور ایده آل، هر نمونه مثال میان افزار IOT باید از سنسورهایی که به طور خاص به آنها متصل است پیگیری شود. این بدان معنی است که هر سرور تنها تعداد مشخصی از سنسورها را می شناسد. با این حال، ممکن است CASSARAM برای پاسخ به نیازهای پیچیده کاربر، به چندین نمونه میان افزار IoT جهت جستجو و انتخاب حسگرهای مناسب مورد نظر نیاز داشته باشد. اجازه دهید سناریوی مربوط به همان حوزه کشاورزی هوشمند را در نظر بگیریم [26]. یک دانشمند می خواهد ببیند که آیا محصولات آزمایشگاهی اش با یک بیماری آلوده است یا خیر. محصولات تجربی او در مناطق پخش شده در مناطق مختلف جغرافیایی استرالیا کاشته می شود. علاوه بر این، حسگرهای مستقر در زمین ها بسته به موقعیت جغرافیایی به نمونه های مختلف میان افزار IoT متصل می شوند. CASSARAM باید به منظور کمک به کاربر برای یافتن سنسورهای مناسب، سرورهای مختلف را به شیوه توزیع شده پرس و جو کند. ما توانایی انجام چنین پرونده های توزیع شده را به خوبی بررسی کردیم. همانطور که در شکل 6 نشان داده شده است، ما سه شیوه مختلف برای شناسایی توزیع کننده های حسگر، بسته به نحوه انتقال داده ها و پرس و جو ها بر روی شبکه (یعنی مسیر)، شناسایی کرده ایم. همچنین نقاط قوت، ضعف ها و کاربرد آنها را در موقعیت های مختلف شناسایی کردیم.

1) پردازش زنجیره: همانطور که در شکل 6 (a) نشان داده می شود، داده ها از یک گره به دیگری به ترتیب ارسال می شود. اول کاربر نیازهای خود را با استفاده از مثال میان افزار IoT تعریف می کند (به عنوان مثال GSN که در یک سرور خاص نصب شده است). سپس، این سرور به عنوان آغازگر درخواست جستجو (SRI) برای این درخواست کاربر مشخص می شود. SRI درخواست را پردازش می کند و 100 حسگر مناسب را انتخاب می کند. سپس، حسگرهای انتخاب شده مربوط به اطلاعات (یعنی شناسه های منحصر به فرد سنسورها و CPWI های مربوطه) به

گره سرور بعدی ارسال می شود. گره دوم (یعنی گره بعدی) اطلاعات حسگر ورودی را با توصیف سنسورهای موجود ادغام می کند و الگوریتم انتخاب سنسور را انتخاب می کند و 100 نوع از بهترین سنسورها را انتخاب می کند. این الگو تا درخواست حسگر از تمام گره های سرور بازدید شده ادامه می یابد. این روش صرفه جویی در پهنای باندی با انتقال ضروری ترین و کمترین مقدار داده ها کاربرد دارد. در مقابل، با توجه به عدم پردازش موازی، زمان پاسخ می تواند بالا باشد.

2) پردازش موازی (همزمان): SRI به طور موازی هر درخواست جستجوی کاربر را به تمام گره های موجود ارسال می کند. سپس هر گره سنسور همزمان یک الگوریتم جستجوی حسگر را انجام می دهد. هر گره 100 حسگر مناسب را انتخاب می کند و حسگرهای انتخاب شده مربوط به اطلاعات را به SRI می رساند. در شرایطی که ما 2500 گره سرور داریم، مقدار اطلاعات (100×2500) که توسط SRI دریافت می شود، می تواند حاکم باشد، به طوریکه پهنای باند ارتباطی را از بین می برد. SRI اطلاعات سنسور را پردازش می کند (100×2500) و 100 حسگر مناسب را انتخاب می کند. وقتی N بزرگتر می شود، این رویکرد ضعیف می شود.

3) پردازش ترکیبی: با مشاهده ویژگی های دو روش قبلی، واضح است که استراتژی پردازش توزیع بهینه باید از تکنیک های پردازش زنجیره ای و موازی استفاده کند. هیچ روش خاصی وجود ندارد که به طور مناسب برای تمام انواع شرایط کاربرد داشته باشد. یک استراتژی پردازش ایده آل توزیع شده برای هر وضعیت نیاز به طراحی دارد و بسته به نوع متن، از قبیل انواع دستگاه ها، توانایی های آنها، پهنای باند موجود و غیره، به صورت پویا تنظیم می شود.

ما می توانیم کارایی روش های فوق را به شرح زیر بهبود بخشیم. همانطور که در شکل 7 (a) نشان داده شده است، در روش پردازش موازی، هر گره، اطلاعات مربوط به سنسور N را به SRI ارسال می کند. با این حال، در پایان، SRI تنها با وجود داشتن مقدار قابل توجهی از اطلاعات مربوط به حسگر ($N \times$ تعداد گره ها) می تواند سنسورهای N را (در مجموع) انتخاب کند. بنابراین، بقیه داده های دریافت شده [$N -$ (تعداد گره ها)] توسط SRI هدر رفته و از بین می رود. به عنوان مثال، فرض کنیم که یک کاربر می خواهد 10,000 سنسور را انتخاب کند. فرض بر این است که 2500 گره سرور وجود دارد، SRI مقدار قابل توجهی از اطلاعات سنسور (10,000 \times 2500) دریافت می کند.

با این حال، ممکن است تنها 10000 سنسور را انتخاب کند. ما روش زیر را برای کاهش این اتلافات پیشنهاد می کنیم، که در شکل 7 (b) نشان داده شده است.

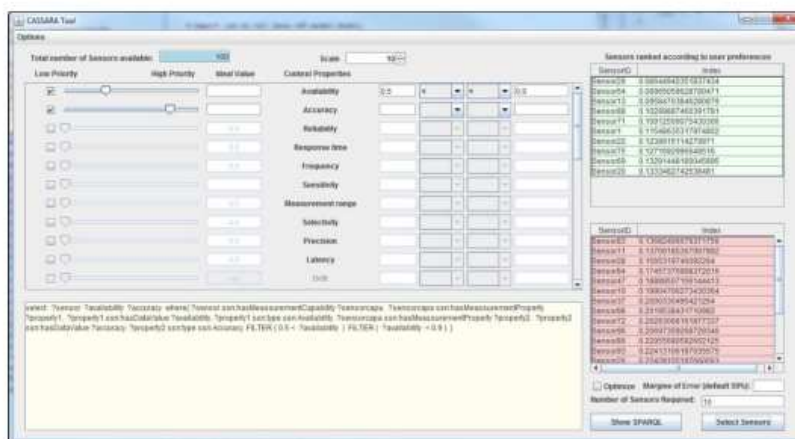
همانطور که در مرحله (1) در شکل b7 نشان داده شده است، در این روش، SRI درخواست جستجو را همزمان یا به طور موازی به هر گره سرور ارسال می کند. هر گره 10000 حسگر مناسب را انتخاب می کند. بدون ارسال اطلاعات در مورد ارسال این 10000 سنسور به SRI، هر گره سرور فقط اطلاعات مربوط به سنسور k_{TH} را ارسال می کند (UID و CPWI هر سنسور k_{TH}). (یعنی، اگر $k = 10000$ سپس گره سرور فقط 1000، 2000، 3000 ... و ده ها هزار سنسور را ارسال می کند). بنابراین، به جای ارسال 10000 ثبت، در حال حاضر هر گره سرور فقط 10 پرونده را باز می گرداند. هنگامی که SRI اطلاعات سنسور را از تمام گره های سرور دریافت می کند، شروع به پردازش می کند و تصمیم می گیرد که کدام قسمت باید بازیابی شود. سپس، SRI درخواست ها را به گره های سرور ارسال می کند و در حال حاضر هر گره بخش دقیق مشخص شده توسط SRI را باز می گرداند (به عنوان مثال، سرور 5 می تواند به جای ارسال 10000 سنسور، فقط به اولین سنسور 2000 بازگردد) همانطور که در (2) نشان داده شده است. در این روش، k نقش کلیدی ایفا می کند و تأثیر مستقیم بر روی کارایی دارد. همانطور که قبلاً ذکر شد، k باید با توجه به N و سایر اطلاعات متنی مربوطه انتخاب شود. به عنوان مثال، اگر ما از k کوچکتر استفاده کنیم، اطلاعات مربوط به سنسورهای بیشتری در مرحله (1) البته با کمترین هزینه در مرحله (2)، به SRI ارسال می شود، در مقابل، اگر ما از k بزرگتر استفاده کنیم، در مرحله (1)، اطلاعات کمتری به SRI ارسال می شود، اما در مرحله (2)، هزینه های نسبتاً بیشتر خواهد بود. علاوه بر این، تکنیک های یادگیری ماشین را می توان بسته به درخواست کاربر و اطلاعات زمینه، مانند انواع سنسورها، انرژی، پهنای باند موجود و غیره برای ارزش گذاری k در هر گره سرور سفارشی داد. مناسب بودن هر رویکرد در قسمت هفتم-B مورد بحث قرار می گیرد.

6. مراحل و تجربیات

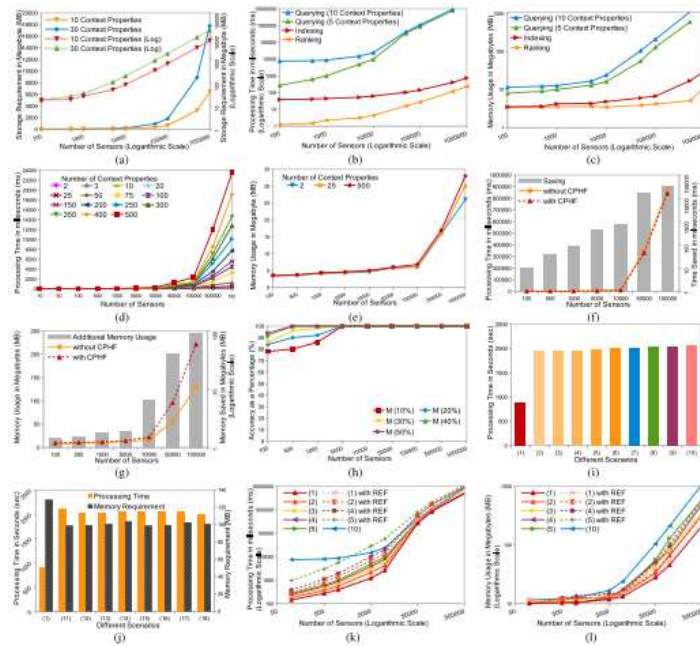
در این بخش، ما تنظیمات تجربی، مجموعه داده های مورد استفاده و پیش فرض ها را توصیف می کنیم. سناریوهای تجربی که ما استفاده کردیم در پایان توضیح داده شده است. بحث در رابطه با آزمایش ها در بخش هفتم ارائه شده

است. Intel(R) Core i5-2557M CPU 1.70GHz و 4 گیگابایت RAM ما مدل پیشنهادی را با استفاده از یک نمونه اولیه، به نام "CASSARA Tool"، که با استفاده از جاوا توسعه دادیم، مورد بررسی و ارزیابی قرار دادیم. رابط کاربر "ابزار CASSARA" در شکل 8 با توصیف خود توضیحی (واضح) ارائه شده است. داده ها در پایگاه داده MySQL ذخیره شدند. ابزار ما اجازه می دهد تا تنظیمات کاربر و اولویت های ویژگی های متنوع سنسورها را بدست آوریم. ما از یک رایانه با CPU Intel (R) Core i5-2557M - 1.70GHz و 4 گیگابایت برای ارزیابی مدل پیشنهادی استفاده کردیم. ما همچنین آزمایشات را با استفاده از رایانه قدرت بالاتر (higher-end computer) با CPU و RAM بیشتر انجام دادیم و نتایج نشان داد که نمودارها شبیه شکل هستند هر چند مقادیر دقیق متفاوت هستند. برای انجام عملیات ریاضی مانند محاسبه فاصله اقلیدس در فضای چند بعدی، ما از کتابخانه عمومی ریاضیات [35] Apache استفاده کردیم. این یک کتابخانه بهینه سازی شده سبک با منبع آزاد است که شامل ریاضیات و اجزای آماری است که به مشکلات شایع غیر قابل دسترس در زبان برنامه نویسی جاوا رسیدگی می کند. همانطور که از هستی شناسی سنسور معنایی (SSN) [5] برای مدیریت توصیف سنسورها و داده های مرتبط استفاده کردیم، ما Apache Jena API منبع باز [36] را برای پردازش و دستکاری داده های معنایی استفاده کردیم. ارزیابی ما با استفاده از ترکیبی از داده های واقعی و داده های مصنوعی تولید شده است. داده های مرتبط با محیط زیست از اداره هواشناسی [37] و مجموعه داده ها از هر دو پروژه Phenonet [27] و پروژه [8] [9] میان افزار مرتبط حسگر (LSM) جمع آوری شد. دلایل اصلی ترکیب داده ها به مقدار زیادی داده و همچنین به کنترل جنبه های مختلف (به عنوان مثال، اطلاعات زمینه مربوط به سنسورهای مورد نیاز برای قرار دادن در مجموعه داده ها، به دلیل داده های واقعی که متناسب با چارچوب زمینه ما هستند در حال حاضر در هیچ مجموعه داده های عمومی در دسترس نیست) جهت درک بهتر رفتار CASSARA در موقعیت ها و سناریوهایی مختلف جهان واقعی IoT که داده های واقعی در آنجا قابل دسترس نیستند، نیاز دارد. ما مفروضات زیر را در کارمان انجام می دهیم. ما فرض می کنیم که توصیف سنسورها و اطلاعات مربوط به سنسورها از تولید کنندگان حسگر از لحاظ هستی شناسی ارزیابی شده اند و به هستی شناسی SSN منتقل شده اند. بدین ترتیب، ما فرض می کنیم که داده های زمینه مرتبط با

سنسورها، مانند دقت، قابلیت اطمینان و غیره، به طور مداوم توسط سیستم های نرم افزاری تحت نظارت، اندازه گیری، مدیریت و ذخیره سازی در هستی شناسی SSN می باشد. ما برای ارزیابی تکنیک های پردازش توزیع شده، یک آزمایش امتحانی شامل چهار گره محاسباتی پیشنهاد کردیم. تمام گره ها به یک شبکه خصوصی سازمانی (به عنوان مثال، IT شبکه ملی دانشگاه استرالیا) متصل می شوند. تنظیمات سخت افزاری سه دستگاه دیگر به شرح زیر است: (1) پردازنده اینتل Core i7 با رم 6 گیگابایت، (2) پردازنده اینتل Core i5 با 4 گیگابایت، و (3) پردازنده اینتل Core i7 با 4 گیگابایت. جزئیات در بخش هفتم-B ارائه شده است.



شکل 8. کاربران در مرحله اول باید ویژگی های متنوعی که به آن وصل هستند را در UI انتخاب کنند. سپس، کاربران باید مقیاس گذاری کنند. هنگامی که مقیاس افزایش می یابد لغزنده ها حساس تر می شوند. در مرحله بعد لغزنده های متصل شده به هر متن باید در الویت قرار گیرند. این مقدار ایده آل مربوط به هر ویژگی متن می تواند ثابت شود. مقادیر می توانند با یک واحد اندازه گیری ابتدایی ثابت شوند (درصد دقت، زمان تاخیر بر حسب یک هزارم ثانیه). تمام مقادیر با CASSARAM عادی شده اند. کاربران با امکان پیش فرض بهترین (به عنوان مثال؛ بیشترین دقت، کمترین هزینه، کمترین تاخیر)، می توانند با انتخاب آن گزینه تصمیم بگیرند که آیا از عملیات بهینه سازی استفاده کنند یا خیر. همچنین کاربران باید حاشیه خطا را به عنوان یک درصد انتخاب کنند (پیش فرض 50٪ است). CASSARAM طبق ارجحیت کاربران SPARQL مناسب را ایجاد می کنند. در نهایت، کاربران باید تعداد حسگرهای مورد نیاز را مشخص کنند.



شکل 9. نتایج آزمایشات CASSARAM. زمان پردازش، کاربرد حافظه، دقت تحت شرایط مختلف (تعنای

حسگرها، تعداد ویژگی های زمینه) به منظور بررسی مطلوبیت و کارآمدی اندازه گیری می شوند.

- (a) تعداد حسگرها (مقیاس لگاریتمی) (b) تعداد حسگرها (مقیاس لگاریتمی) (c) تعداد حسگرها (مقیاس لگاریتمی) (d) تعداد حسگرها (e) تعداد حسگرها (f) تعداد حسگرها (g) تعداد حسگرها (h) تعداد حسگرها (i) سناریوهای مختلف (j) سناریوهای مختلف (k) تعداد حسگرها (مقیاس لگاریتمی) (l) تعداد حسگرها (مقیاس لگاریتمی)

ما عملکرد CASSARAM را با استفاده از ترکیبات مختلف اپراتورهای ارتباطی مانند $<$ ، $>$ ، $=$ ، \geq ، \leq ارزیابی کردیم. سناریوهای موجود در اشکال 9i-9l مربوط به شماره های سناریو ذکر شده در زیر است. تمام آزمایشات پنج ویژگی خاص را بازیابی می کند. (1) از اپراتور ارتباطی استفاده نکنید. (2) 1 ویژگی از 5 ویژگی زمینه با \leq محدود شده است (به عنوان مثال، دقت بیش از 80٪). (3) 2 ویژگی از 5 ویژگی (به عنوان مثال، دقت بیش از 80٪ و قابلیت اطمینان بیش از 85٪). (4) 3 ویژگی از 5 ویژگی، (5) 4 ویژگی از 5 ویژگی. همه 5 ویژگی (6) توسط \leq ، (7) توسط \geq (8) توسط $=$ ، (9) توسط $>$ ، (10) توسط $<$ محدود شدند. (11) 1 ویژگی از 5 ویژگی زمینه توسط دو اپراتور ارتباطی محدود شده است (به عنوان مثال، دقت بیشتر از $\leq 80\%$ و کمتر از 95% است)، (12) 2 ویژگی از 5 ویژگی، (13)

3 ویژگی از 5 ویژگی، (14) 4 ویژگی از 5 ویژگی؛ همه 5 خصوصیات زمینه توسط \geq و \leq ، (16) توسط $<$ و $>$ محدود می شوند (15). ما تعدادی محدودیت اعمال شده را با استفاده از اپراتورهای ارتباطی بیشتر افزایش دادیم. (17) دو محدوده را برای هر یک از خصوصیات زمینه تعریف می کند (به عنوان مثال (دقت $\leq 80\%$ و $\geq 95\%$) OR (دقت $\leq 50\%$ و $\geq 60\%$)). (18) سه دامنه را تعریف می کند.

7. ارزیابی و بحث

ما با استفاده از روش ها و پارامترهای مختلف که در اشکال نشان داده شده است، CASSARAM ارزیابی کردیم. 9i-9I ما در این بخش معیارهای ارزیابی را برای هر آزمایش استفاده می کنیم و همچنین در مورد درس هایی که یاد گرفتیم، توضیح می دهیم. شکل a9 نشان می دهد که نیاز ذخیره سازی بسته به تعداد توصیف های حسگر متفاوت است. همانطور که در شکل 3 نشان داده شده است، ما داده ها را با توجه به هستی شناسی SSN ذخیره کردیم. ما دو آزمایش انجام دادیم که در آن 10 خواص زمینه و 30 ویژگی زمینه را از چارچوب زمینه ای که در بخش چهارم-F پیشنهاد شده، ذخیره کردیم. برای ذخیره توصیف یک میلیون سنسور، 6.4 گیگابایت (10 ویژگی زمینه) و 17.8 گیگابایت (30 ویژگی محدوده) به کار بردیم. واضح است که نیازهای ذخیره سازی با تعداد سه برابر همبستگی دارند: یک بار نیاز سه گانه در حدود 0.193 کیلو بایت حافظه ذخیره سازی (برای 100,000+ سنسور) دارد. اگر چه سخت افزار ذخیره سازی در حال ارزانتر شدن است و در ظرفیت های بالا در دسترس است، تعداد خواص زمینه برای ذخیره باید با دقت تصمیم گیری شود تا نیازهای ذخیره سازی به حداقل برسد، به ویژه وقتی که ارزش تعداد سنسورها میلیاردها دلار باشد. شکل b9 نشان می دهد چه مقدار زمان برای انتخاب سنسور ها به عنوان تعداد سنسور ها افزایش می یابد. هر مرحله (یعنی جستجو، شاخص گذاری (نمایه سازی) و رتبه بندی) به طور جداگانه اندازه گیری شده است. پرس و جو معنایی نسبت به شاخص گذاری و رتبه بندی به زمان بیشتر برای پردازش نیاز دارد. علاوه بر این، از آنجائیکه تعدادی از ویژگی های زمینه ارزیابی شده توسط پرس و جو افزایش می یابد، زمان اجرا نیز به طور قابل توجهی افزایش می یابد. علاوه بر این، مهم است که توجه داشته باشید که MySQL می تواند تنها 61 جدول داشته باشد که فقط اجازه می دهد حداکثر 10 ویژگی زمینه را از مدل داده های هستی شناسی SSN ارزیابی کند. استفاده

از ذخیره سازی داده های جایگزین یا اجرای چندین نمایش داده شده می تواند برای غلبه بر این مشکل مورد استفاده قرار گیرد. بدین ترتیب، در صورتی که تعداد سنسورها کمتر از 10,000 باشد (مثلا 8 میلی ثانیه برای بازیابی 5 خواص کانتینر و 24 میلی ثانیه برای بازیابی 10 ویژگی زمینه در هنگام پرسیدن 10,000 سنسور)، اجرای نمایشهای چندگانه از اجرای یک پرس و جو بسیار کارآمدتر است. علاوه بر این، شکل c9 نشان می دهد که مقدار حافظه برای انتخاب سنسور ها مانند تعداد سنسور ها افزایش می یابد. واضح است که داشتن ویژگی های زمینه به داشتن حافظه بیشتری نیاز دارد. الزامات حافظه مختص پرس و جو بیش از 10,000 تغییر نمی کند (از 10 مگابایت تا 25 مگابایت). هنگامی که تعداد سنسورها بیش از 10,000 است، نیازهای حافظه به طور پیوسته رشد می کنند و با تعداد سنسورها ارتباط دارد. که در مقایسه، شاخص گذاری و رتبه بندی نیاز به حافظه کمتری دارد.

شکل d9 نشان می دهد که زمان پردازش صرف شده در فرایند شاخص گذاری حسگر به عنوان تعدادی از ویژگی های زمینه و تعداد سنسور ها افزایش می یابد. کاهش تعداد سنسورهایی که نیاز به اندیس کمتر از 10000 دارند اجازه می دهد سرعت CASSARAM افزایش یابد. زمان پردازش پس از 100,000 سنسور به طور قابل توجهی شروع به افزایش می کند. به طور مشابه، شکل e9 استفاده از حافظه (memory usage) توسط فرایند نمایه سازی حسگر را به عنوان تعدادی از ویژگی های زمینه و سنسور افزایش می دهد. حتی اگر نیازهای حافظه کمی افزایش یابد، وقتی که تعداد سنسورها هنوز کمتر از 100000 است، افزایش واقعی ناچیز است. پس از آن، الزامات حافظه به طور قابل ملاحظه ای افزایش می یابد اما در مقایسه با قابلیت های محاسباتی جدیدترین سخت افزار هنوز هم بسیار کوچک است. علاوه بر این، تعدادی از خواص زمینه درگیر تاثیر قابل توجهی در روند شاخص گذاری نداشته است. زمانی که تعداد سنسورها به یک میلیون رسانده می شود، تفاوت قابل مشاهده است. با این حال، حافظه مورد نیاز این پردازنده 30 مگابایت است. جمع آوری زباله جاوا، هنگام پردازش تعداد زیادی از سنسورها، کار خود را به طور فعال انجام می دهد، که این تفاوت را غیر قابل مشاهده می کند.

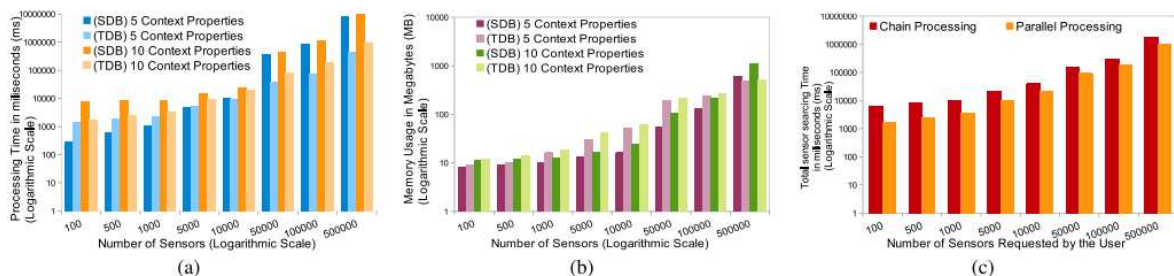
در شکل f9 و g9 زمان صرف شده در فرآیند انتخاب سنسور و حافظه ای که نیاز دارد، با و بدون الگوریتم CPHF با افزایش تعداد سنسور ها مقایسه می شود. تعداد سنسورهایی که نیاز به کاربر دارد در 50 آزمایش از بین همه

آزمایشها ($N=50$) نگهداری می شود. پنج ویژگی بازیابی، نمایه سازی (شاخص گذاری) و رتبه بندی می شوند. پیچیدگی CPHF (به دلیل زیرشاخه SPARQL) تاثیر قابل توجهی بر کل زمان پردازش CASSARAM نداشته است. در عوض، CPHF برخی از زمان ها را در مرحله های نمایه سازی و رتبه بندی ذخیره کرده است. در مقابل CPHF به دلیل پیچیدگی نیاز به حافظه بیشتری دارد. با این حال، هنگام انتقال داده ها به مرحله بعدی برای نمایه سازی، به میزان قابل توجهی کمتر به حافظه نیاز دارد. بنابراین، از آنجائیکه CPHF نیاز به نگهداری میلیون ها قطعه اطلاعات حسگر در مراحل مختلف در CASSARAM ندارد، کارآیی دارد. علاوه بر این، CPHF تنها تعداد محدودی از سنسورها را باز می گرداند، در حالی که روش غیر CPHF تمامی سنسورهای موجود در CASSARAM را باز می گرداند که منابع بیشتری از جمله زمان پردازش بیشتر و میزان قابل توجه حافظه و ذخیره سازی موقت را صرف می کنند. شکل h9 نشان می دهد که چگونه دقت زمانی تغییر می کند که مقدار ($M\%$) حاشیه خطا (Margin of Error) در الگوریتم CPHF تغییر می کند و تعداد سنسور ها افزایش می یابد. سناریو ارائه شده در شکل 5 ارزیابی شده است. دقت روش رویه CPHF هنگامی که حاشیه خطا (M) افزایش می یابد، افزایش می یابد. با این حال، M پایین تر CASSRAM را به سمت مصرف کم منابع هدایت می کند. بنابراین، بین دقت و مصرف منابع، اختلاف وجود دارد. ارزش بهینه M می تواند به صورت پویا با استفاده از تکنیک های یادگیری ماشینی بر اساس ویژگی های زمینه ای اولویت بندی شده توسط کاربر در هر وضعیت و چگونگی وزن های نرمال شده توزیعی در بین خواص متفاوت آموخته شود.

در شکل i9 و شکل j9، ما بررسی کردیم که زمان پردازش و نیازهای حافظه زمانی تغییر می کند که اصطلاحات رابطه ای در طی مرحله پرس و جو معنایی استفاده می شود. ما سناریوهای مختلف را با و بدون اصطلاحات ارتباطی (مثلا $<$ ، $>$ ، $=$ ، \geq ، \leq) آزمایش کردیم که در انتهای بخش پنجم توضیح داده شد. برای تمام آزمایشات، ما 100000 سنسور را مورد پرسش قرار دادیم. هنگامی که حداقل یک اپراتور ارتباط در SPARQL استفاده می شود، زمان پردازش و نیازهای حافظه 100٪ افزایش می یابد. با این حال، نه تعداد اپراتورهای ارتباطی و نه نوع اپراتورهای ارتباطی استفاده شده هیچ تأثیری بر زمان پردازش یا نیاز به حافظه ندارد. بنابراین، استفاده از چندین اپراتور ارتباطی (تا آنجا که ممکن

است) مفید است، به طوری که تعداد سنسورهای بازیابی شده توسط فاز پرس و جو کاهش می یابد. این امر کمک می کند تا میزان داده هایی که در مراحل دیگر نیاز به پردازش دارند را کاهش دهد.

در نهایت، در شکل K9 و L9، ما به طور گسترده ای بررسی کردیم که چگونه REF بر زمان پردازش و نیازهای حافظه در CASSARAM تأثیر می گذارد، به طوری که تعداد سنسورها و خصوصیات زمینه افزایش می یابد. همانطور که قبلاً ذکر شد، REF بیشتر هزینه های پردازش را اضافه می کند که بر زمان پردازش و حافظه تأثیر می گذارد. زمانیکه تعداد سنسورهایی که نیاز به درخواست دارند کمتر از 100,000 است، تفاوت معنی داری در زمان پردازش وجود دارد. با این حال، هنگامی که تعداد سنسورها بیش از 100,000 افزایش می یابد، تفاوت این امر ناخوشایند می شود. در مقابل، اختلاف در نیازهای حافظه ناچیز است، زیرا تعداد سنسورها کمتر از 10,000 است، اما پس از آن شروع به تبدیل شدن قابل مشاهده است. علاوه بر این، زمان پردازش به طور قابل توجهی پس از 10,000 سنسور افزایش می یابد. ما همچنین یاد گرفتیم که اختصاص حافظه بیشتر به CASSARAM می تواند سرعت کل فرایند انتخاب سنسور را افزایش دهد.



شکل 10. نتایج استفاده از ذخیره سازی جایگزین و جستجوی حسگرهای توزیع شده: (a) مقایسه زمان پردازشی هر دو روش Jena SDB / MySQL و LTE Jena؛ (b) مقایسه استفاده از حافظه با روشهای SDB و TDB؛ و (c) مقایسه زمانی که دو روش مختلف جستجوی توزیع انجام می شود، یعنی پردازش زنجیره ای و پردازش موازی

- (a) تعداد حسگرها (مقیاس لگاریتمی) (b) تعداد حسگرها (مقیاس لگاریتمی) (c) تعداد حسگرهای درخواست

شده توسط کاربر

در مقابل، CASSARAM نیز می تواند در منابع محدود استفاده شود، گرچه زمان بیشتری طول می کشد تا پاسخ دهد. با توجه به ارزیابی های گسترده انجام شده، مشخص است که تکنیک های CPHF و REF می تواند برای بهبود کارایی CASSARAM استفاده شود. اگر چه این مقاله به طور خاص بر روی انتخاب سنسور در دامنه IoT متمرکز شده است، مدل پیشنهادی و مفاهیمی که ما استفاده می کنیم می تواند در بسیاری از زمینه های دیگر مانند انتخاب خدمات وب مورد استفاده قرار گیرد. علاوه بر این، نتایج به دست آمده از طریق این ارزیابی ها نیز برای هر رویکرد دیگری که از یک مدل هستی شناسی شبیه به هستی شناسی SSN استفاده می کند و به تعداد زیادی سوابق نیاز دارد، قابل استفاده است. با وجود اینکه ما راه حل را با میلیون ها توصیف حسگر مورد آزمایش قرار دادیم، در عمل بسیار بعید است که میلیون ها حسگر به نمونه یک میان افزار خاص وصل شوند. در واقع، راه حل های میان افزار IoT داده ها را در شیوه های توزیع شده در موارد مختلف ذخیره می کنند و باید به صورت توزیعی مورد جستجو قرار گیرند، همانطور که در بخش پنجم-C توضیح داده شده است، می توان با پردازش موازی، مقدار زمان لازم برای پردازش میلیون ها توصیف داده های سنسور را به شدت کاهش داد.

A. ارزیابی گزینه های ذخیره سازی جایگزین

ما در ارزیابی های انجام شده در ابتدا (شکل 9-19a)، از ذخیره سازی RDF با پشتیبانی Jena SDB / MySQL استفاده کردیم. ما در اینجا به هنگام استفاده از گزینه های ذخیره سازی جایگزین جهت ذخیره داده ها از یک روش پشتیبان Jena TDB (jena.apache.org/documentation/tdb) برای ارزیابی عملکرد CASSARAM استفاده می کنیم. ما در شکل 10c، زمان پردازش را با استفاده از روش Jena SDB / MySQL و Jena TDB مقایسه می کنیم. علاوه بر این، در شکل 10b ما استفاده از حافظه را با روش های SDB و TDB مقایسه می کنیم. بر اساس معیار (benchmark) SPARQL برلین [30] مشخص شد که Jena TDB بسیار سریعتر از Jena SDB است. ما همچنین نتایج مشابهی را در 5 پردازش داده های متنی و همچنین در 10 پردازش داده های متنی مشاهده کردیم. بخصوص اینکه، Jena TDB هنگام پردازش 10 خواص متنی که در آن مجموعه داده ها شامل نیمی از میلیون توصیف سنسور است، 10 برابر سریعتر از SDB است. زمانی که مجموعه داده کمتر از 100,000 توصیف

حسگر بود، رویکرد JEN SDB، حافظه کمتری نسبت به روش Jena TDB صرف کرد. با این حال، پس از آن، رویکرد Jena TDB حافظه کمتری نسبت به SDA Jena مصرف می کند. بویژه اینکه، Jena TDB هنگام پردازش 10 ویژگی متنی، که مجموعه داده شامل نیمی از میلیون توصیف حسگر را در بر می گیرد، نسبت به Jena SDB، کمتر از 50٪ از حافظه استفاده می کند. بنابراین واضح است که Jena TDB مناسب تر است زیرا تعداد توصیف های حسگر فراتر از 100,000 فراتر می رود.

با وجود تفاوت هایی که در ارزیابی ما مشاهده کردیم، عوامل متعددی وجود دارد که باید هنگام انتخاب راه حل های ذخیره سازی زیر مورد توجه قرار گیرد. همانطور که در معیار برلین SPARQL ارزیابی می شود، چندین گزینه ذخیره سازی دیگر مانند Sesame (openrdf.org)، Virtuoso TS، Virtuoso RV و Server R2D [30] وجود دارد. Jena TDB سرعت بار بارگیری و مقیاس بهتر را ارائه می دهد، اما بدترین عملکرد پرس و جو را دارد. به نظر می رسد Sesame با توجه به بارهای نامنظم برای اندازه های کم بهتر باشد. در مقابل، Jena SDB عملکرد متوسط، زمان ارائه بار، عملکرد پرس و جو و مقیاس پذیری بین Jena TDB و Sesame را فراهم می کند. بر اساس این ارزیابی ها، زمانی که این مقاله نوشته شده است، هیچ راه حلی برتر وجود ندارد که دارای ویژگی های خوب باشد. با توجه به عدم استفاده گسترده و وجود کوتاه Sesame، SDB / MySQL را می توان به عنوان یک انتخاب بهتر، به ویژه هنگام استفاده از ویژگی های پایگاه داده مانند پشتیبان گیری (backup)، پردازش همزمان و موازی، در نظر گرفت. همانطور که بارگیری مکرر یا تخلیه داده های مجموعه ای مانند توصیف های حسگر را انتظار نداریم، بدیهی است SDB در پردازش پرس و جو (به استثنای بارگیری داده ها) بهتر از TDB عمل می کند [30]. همانطور که انتظار می رود SDB برای به روز رسانی بیشتر (معاملات)، انتخاب خوبی خواهد بود.

B. ارزیابی جستجوی حسگر توزیع شده

ما با استفاده از یک شبکه خصوصی که شامل چهار گره محاسباتی است، جستجوی حسگر توزیع شده را ارزیابی کردیم. ما دو روش مختلف جستجوی حسگر توزیع را مقایسه می کنیم، یعنی پردازش زنجیره ای و پردازش موازی با یا بدون انبساط یا گسترش k که ما در بخش پنجم - C بحث کردیم. نتایج در شکل c10 ارائه شده است. هر گره شامل یک

مجموعه داده حاصل از یک میلیون توضیحات داده سنسور است. چهار مجموعه داده با یکدیگر فرق دارند. پنج خصوصیت زمینه برای ارزیابی در نظر گرفته شده و اطلاعات متنی با استفاده از Jena TDB ذخیره می شود. ابتدا ما در مورد تکنیک های دیدگاه نظری بحث می کنیم.

اجازه دهید برخی از عبارات را که در بحث زیر مورد استفاده قرار می گیرند را تعریف کنیم: n = تعداد گره های محاسباتی (در آزمایشات ما $n = 4$)، N = تعداد سنسورهای درخواست شده توسط کاربران، S_i = تعداد توصیف های حسگر ذخیره شده در گره محاسباتی i ، r = اندازه یک سابقه توصیف سنسور (یعنی نیازهای ذخیره سازی)، $t_{i,j}^{net}$ = زمان برای ارتباط شبکه بین گره های محاسباتی i و t_i^{pro} = زمان گرفته شده برای پرس و جو از گره محاسباتی i ، ادغام نتایج نشان داده شده با نتایج ورودی و تعداد نهایی حسگرها را انتخاب کنید. زمان کل انجام شده توسط جستجوی زنجیره توزیع سنسور می تواند تعیین شود:

$$Total_{chain} = \sum_{i=1}^n t_i^{pro} + \sum_{i=1}^{n-1} t_{i,i+1}^{net} + t_{n,1}^{net}. \quad (1)$$

کل زمان صرف شده در جستجوی حسگر توزیع موازی می تواند به شکل زیر تعریف شود:

$$Total_{parallel} = \max \{i = [2..n] : t_i^{pro} + t_{1,i}^{net}\}. \quad (2)$$

طبق نتایج، مشخص است که از نظر کل زمان پردازش، پردازش موازی بهتر از پردازش زنجیره ای است. با این حال، پردازش موازی در سایر جنبه ها، از قبیل ارتباطات شبکه و مصرف پهنای باند، ناکارآمد است. بنابراین، ما گسترش K - را برای این مسئله پیشنهاد کردیم. ارزیابی روش گسترش k - در جدول 3 ارائه شده است. در این آزمایش، ما اندازه گیری کردیم که چگونه می توان ارتباطات داده ها را ذخیره کرد (به عنوان مثال، به دلیل حذف ارتباط اضافی² که در پردازش موازی بدون گسترش k رخ می دهد) با استفاده از مقادیر k مختلف با مقادیر مختلف N . ما حداقل حداقل 2 نوع ارتباطات داده را اندازه گیری کردیم (اندازه گیری شده در مگابایت) که می تواند ذخیره شود.

² پردازش موازی با گسترش K بسته به مجموعه داده ها و اطلاعات متنی ذخیره شده در هر گره، قادر خواهد بود ارتباطات داده ای بیشتری را از سطح حداقل تضمین شده ذخیره کند.

در جدول 3، مقادیر مثبت (نشان داده شده در سبز) نشان می دهد حداقل ارتباط داده ها با استفاده از گسترش-K ذخیره شده است. گرچه مقادیر منفی (مشخص شده با نارنجی / قرمز) نشان دهنده عدم پس انداز تضمین شده است، بعضی از موقعیت ها (مشخص شده با نارنجی) در مقایسه با دیگران شانس بیشتری برای صرفه جویی داده ها دارند. معادله (3) را می توان با استفاده از گسترش-k برای محاسبه حداقل مقدار تضمین صرفه جویی استفاده کرد.

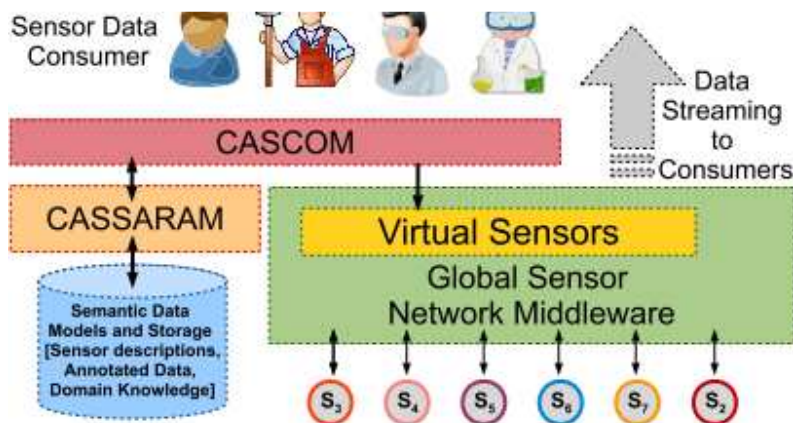
$$Total_{Saving} = \sum_{i=2}^n S_i r - \left\{ \left[\sum_{i=2}^n \frac{S_i}{k} + N + (k-1)n \right] \times r \right\},$$

IF ($k < N$). (3)

جدول 3: مقدار ارتباطات تکراری داده ذخیره شده توسط تحقیق همزمان حسگرها با روش توسعه K

k value	Number of sensors requested by the users (N)								
	100	500	1,000	5,000	10,000	50,000	100,000	500,000	1,000,000
10	-60.7	-60.5	-60.3	-58.7	-56.7	-40.5	-20.2	141.6	344.0
100		-5.9	-5.7	-4.1	-2.1	14.1	34.3	196.2	398.5
500			-1.1	0.5	2.5	18.7	38.9	200.8	403.1
1000				0.8	2.8	19.0	39.3	201.1	403.5
5000					0.9	17.1	37.3	199.2	401.5
10000						14.1	34.3	196.2	398.5
50000							10.1	172.0	374.3
100000								141.6	344.0
500000									101.2

in Megabytes (MB)



شکل 11: CASSARAM در عمل.

اجازه دهید صحنه های مختلفی را که در آن پردازش زنجیره ای و موازی می تواند مورد استفاده قرار گیرد را در نظر بگیریم. پردازش زنجیره ای برای شرایطی مناسب است که صرفه جویی در منابع محاسباتی و پهنای باند مهم تر از

زمان پاسخ است. روش پردازش موازی بدون گسترش k - زمانی مناسب است که زمان پاسخ بحرانی و N نسبتاً کوچک است. گسترش K - نیاز به دو دور ارتباط دارد: رادیوهای ارتباطی باید دو بار باز و بسته شوند. چنین الگوی ارتباطی انرژی بیشتری مصرف می کند [38]، به ویژه اگر دستگاه های محاسباتی انرژی محدود باشند. بنابراین، انتقال داده ها وقتی که یک بار انجام شود، کارآیی بیشتری دارد. با این حال، این توصیه زمانی که N به شدت بزرگ می شود (+000,10) نامعتبر می شود. آزمایشات ما به وضوح نشان می دهد که گسترش k - می تواند برای بهبود کارایی روش جستجوی حسگر موازی، به ویژه هنگامی که N بزرگ است، استفاده شود. ارزش ایده آل k باید بر اساس N ، n و S_i تعیین شود.

C. برنامه کاربردی

در این بخش، ما نشان می دهیم که CASSARAM در تصویر بزرگ نشان داده شده است (شکل 11). انتظار می رود مصرف کنندگان یا کاربران داده های سنسور با یک مدل به نام مدل پیکربندی حسگر زمینه آگاهانه (CASCoM) ارتباط برقرار کنند [39]. شرح مفصل CASCoM خارج از محدوده این مقاله است. مصرف کنندگان با یک رابط کاربری گرافیکی که براساس یک رویکرد پرسش-پاسخ (QA) است که اجازه می دهد تا شرایط را بیان کند، تسهیل می شوند. کاربران می توانند به عنوان بسیاری از سوالات در صورت امکان جواب دهند. CASCoM جستجو می کند و وظایفی را که می تواند انجام دهد را بررسی می کند. کاربران می توانند کار مورد نظر را از لیست فیلتر شده انتخاب کنند (به عنوان مثال تشخیص آلودگی محیط زیست). CASCoM برای اجزای برنامه نویسی مختلفی که امکان ضبط جریان داده مورد نیاز مصرف کنندگان را فراهم می کند، جستجو کند. (به عنوان مثال داده های حسگر مورد نیاز برای تشخیص آلودگی محیط زیست). CASCoM تلاش می کند تا حسگرهایی را پیدا کند که می توانند برای تولید ورودی های مورد نیاز توسط اجزای پردازش داده ها مورد استفاده قرار گیرند. CASSARAM برای رسیدن به این مرحله کاری از CASCoM استفاده می کند. هنگامی که انواع سنسورهای مورد نیاز شناسایی می شوند (و اگر چند سنسور در دسترس باشند)، رابط کاربری گرافیکی CASSARM برای تعیین اولویت های مصرف کنندگان به آنها ارائه می شود. در مرحله بعد، مجموعه ای از آخرین سنسورها و اجزای پردازش اطلاعات با هم ترکیب می شوند.

پیچ و تاب های مورد نیاز [40] و حسگر مجازی [3] تولید می شوند و توسط CASCOM به GSN ارسال می شوند. در نهایت، همانطور که در سنسور مجازی مشخص شده است GSN شروع به جریان انداختن داده ها برای مصرف کننده می کند.

8. نتیجه گیری ها و پژوهش های آینده

با پیشرفت تکنولوژی سخت افزار سنسور و مواد ارزان، انتظار می رود حسگرها به تمام اشیاء اطراف ما متصل شوند، که باعث افزایش تعداد سنسورهای موجود برای استفاده می شود. این به این معنی است که ما به سنسورهای متعدد دسترسی داریم که پدیده زیست محیطی مشابه را اندازه گیری می کنند. چنین شرایطی ما را مجبور می کند که از بین گزینه ها انتخاب کنیم. ما باید تصمیم بگیریم که کدام ویژگی های متنی مرتبط با سنسور عملیاتی و مفهومی مهم تر از بقیه است.

در این مقاله، ما نشان دادیم که چگونه اطلاعات متنی مربوط به هر سنسور می تواند برای جستجو و انتخاب سنسورهایی که به بهترین وجه مورد نیاز کاربر هستند، مورد استفاده قرار گیرد. ما سنسورها را براساس انتظارات و اولویت های کاربر انتخاب کردیم. ما به عنوان اثبات یک مفهوم، نمونه اولیه ای با استفاده از برنامه های واقع گرایانه برای نشان دادن عملکرد CASSARAM و حمایت از آزمایش ها ساختیم. ما همچنین نشان می دهیم که چگونه CASSARAM جهت رسیدن به دیدگاه وسیع تر با چشم انداز وسیع تر به عنوان یک سرویس در پارامتر IoT. سنجش از دید شما در پارادایم IoT کمک می کند. CASSARAM بهینه سازی روش های جمع آوری داده های سنسور را با انتخاب سنسورها به صورت بهینه انجام می دهد. به عنوان مثال، CASSARAM می تواند مورد استفاده قرار گیرد تا دریابند که کدام حسگرها انرژی بیشتری دارند و فقط از آن سنسورها اطلاعات را جمع آوری می کنند. این کمک می کند تا کل شبکه حسگر را برای مدت زمان بسیار طولانی بدون نیاز به پیکربندی مجدد اجرا کنید. ما سه تکنیک مختلف را که بهبود و مقیاس پذیری CASSARAM را بهتر می کند، بررسی کردیم که شامل: فیلترینگ اکتشافی مبتنی بر الویت تطبیقی، فیلترینگ مبتنی بر بیان رابطه، و جستجوی حسگر توزیع شده است. ما عملکرد مدل پیشنهادی را به طور گسترده مورد ارزیابی قرار دادیم. در آینده ما قصد داریم CASSARAM را به راه حل

های پیشرو میان افزار IOT مانند GSN، SenseMA و OpenIoT و بسیاری تا از عملکرد انتخاب خودکار سنسور در محیط های توزیع پشتیبانی شود. ما همچنین بررسی چگونگی بهبود کارایی CASSARAM با استفاده از جستجوی حسگر مبتنی بر خوشه و الگوریتم های اکتشافی که شامل تکنیک های یادگیری ماشین می شوند، را بررسی خواهیم کرد.

سیاسگذاری

نویسندگان از کمک و همکاری دانشگاه ملی استرالیا، کانبرا، و استرالیا قدردانی می کنند.

منابع

- [1] C. Perera D. Georgakopoulos, A. Zaslavsky, "دریافت کردن به عنوان یک سرویس و داده بزرگ" در *Proc. Int. Conf. ACC*, Jul. 2012, pp. 21–29.
- [2] (2012) راه حل منبع باز برای اینترنت اشیا درون ابر [آنلاین]. در دسترس: <http://www.openiot.eu>
- [3] A. Salehi و M. Hauswirth, K. Aberer "زیرساخت برای پردازش داده ها در شبکه های حسگر متصل شده در مقیاس بزرگ،" در *Proc. Int. Conf.* مدیریت داده موبایل، pp. 198–205, May 2007.
- [4] C. P. Elzaker, T. Bartoschek, F. Bache, A. Broring, "یک رویکرد بصری برای ادغام سنسورها با وب سنسور، در جهت پیشبرد علم اطلاعات جغرافیایی برای یک دنیای در حال تغییر،" S. Geertman, Ed. F. Toppen, W. Reinhardt, نیویورک، نیویورک، ایالات متحده آمریکا: Springer-Verlag, 2011, pp. 143-162.
- [5] (ژوئن 2011). گزارش نهایی شبکه حسگر معنایی XG: گزارش گروهی انکوباتور W3C <http://www.w3.org/2005/Incubator/ssn/XGR-ssn-20110628/>
- [6] D. Georgakopoulos, P. Christen, A. Zaslavsky, C. Perera "زمینه محاسبات هوشمندانه برای اینترنت اشیا: نظرسنجی،" *IEEE Commun. نظرسنجی*, Tuts, vol. PP, no. 99, pp. 1–41, May 2013.

- [7] P. Barnaghi, W. Wang, C. Henson, و K. Taylor, معنا شناسی اینترنت اشیا: پیشرفت های اولیه و بازگشت به آینده، *Int. J.* وب معنایی *Inf. Syst.*, vol. 8, no. 1, pp. 1–21, Jan. 2012
- [8] موسسه تحقیقاتی دیجیتال، (2011). میان افزار سنسور متصل شده (Galway (LSM), ایرلند [آنلاین]. موجود: <http://lsm.derl.ie/>
- [9] D. L. Phuoc, H. N. M. Quoc, J. X. Parreira, و M. Hauswirth, میان افزار سنسور متصل شده، متصل کردن دنیای واقعی و وب معنایی "، در *Proc. ISWC*, Oct. 2011, pp. 1–22.
- [10] S. Nath, J. Liu, و F. Zhao, نقشه حسگرانه برای شبکه های حسگر گسترده، کامپیوتر، vol. 40, no. 7, pp. 90–93, Jul. 2007
- [11] C. Perera, A. Zaslavsky, P. Christen, M. Compton, D. Georgakopoulos, جستجو، انتخاب و مدل رتبه بندی برای جستجوگر حساس به متن برای اینترنت میان افزاری اشیا، " in *Proc. IEEE 14th Int. Conf. MDM*, Jun. 2013, pp. 314–322.
- [12] C. Truong, K. Romer, و K. Chen, "جستجوی حسگر فازی- محور در وب اشیا" در pp. 127–134. *Proc. 3rd Int. Conf. IoT 2012*, 2012, . .
- [13] S. Mayer, D. Guinard, و V. Trifa, جستجو در وب زیرساخت-محور برای اشیای هوشمند "در *Proc. 3rd Int. Conf. Internet Things*, 2012, pp. 119–126
- [14] B. M. Elahi, K. Romer, B. Ostermaier, M. Fahrmaier, و W. Kelleher, رتبه بندی سنسور: جستجوی ابتدایی برای جستجوی حسگرهای محتوا محور کارآمد، *Proc. Int. Conf. Inf. Process. Sensor Netw.* 2009pp. 217–228,
- [15] B. Ostermaier, K. Roalter, L. Oandmer, F. Mattern, M. Fahrmaier, و W. Kellerer, موتور جستجو بلادرنگ برای اینترنت اشیا"، در *Proc 2nd Int. Conf. IoT*, Dec. 2010, pp. 1–8.
- [16] J. Domingue و D. Fensel, گرایش به یک سرویس وب: ادغام وب معنایی و جهت گیری خدماتی، *IEEE Intell. Syst.*, vol. 23, no. 1. pp. 86–88, Dec. 2009.

- [17] S. De, T. Elsaleh, P. Barnaghi, S. Meissner, "اینترنت پلت فرم اشیا برای اشیاء دنیای واقعی و دیجیتال،" محاسبات مقیاس پذیر. *Pract. Expert.* vol. 13, no. 1, pp. 45–57, 2012.
- [18] D. Guinard, V. Trifa, S. Karnouskos, P. Spiess, D. Savio, "تعامل با اینترنت اشیا مبتنی بر SOA: کشف، پرس و جو، انتخاب، و ارائه تقاضا محور خدمات وب،" *IEEE Trans.* خدمات محاسبات، vol. 3, no. 3, pp. 223–235, Jul. /Sep. 2010
- [19] T. A. Butt, J. Phillips, L. Guan, G. Oikonomou, "گرایش: یک پروتکل کشف خدمات تطبیقی و متمرکز برای 6LoWPANs," *Proc. 3rd Int.* وب اشیا کارگاهی، 2012, pp. 2:1–2:6.
- [20] Z. Shelby, "خدمات وب تعبیه شده"، ارتباطات بی سیم. *IEEE*, vol. 17, Dec. 2010, pp. 52–57, no. 6, 2010
- [21] J.-P. Calbimonte, H. Jeung, O. Corcho, K. Aberer, "فعال سازی فن آوری های پرس و جو برای وب سایت سنسور معنایی،" *Int. J. Semant. Web Inf. Syst.*, vol. 8, no. 1, pp. 43–63, Jan. 2012
- [22] R. Garcia-Castro, O. Corcho, C. Hill, "یک مدل هستی شناسانه هسته ای برای زیرساخت های وب سنسور،" *Int. J. Semantic Web Inf. Syst.*, vol. 8, no. 1, pp. 22–42, Jan. 2012
- [23] N. Namatame, Y. Ding, T. Riedel, H. Tokuda, T. Miyaki, M. Beigl, معماری مدیریت منابع توزیع شده برای اتصال وب سایت های استفاده کننده از uBox، *Proc. 2nd Int.* اشیا وب کارگاهی 2011, pp. 4:1–4:6.
- [24] J.-P. Calbimonte, H. Jeung, O. Corcho, K. Aberer, جستجوی داده حسگر معنایی در یک شبکه حسگر تشکیل شده در مقیاس بزرگ،" *Proc. 4th Int.* شبکه سنسور مفهومی کارگاهی، Oct. 2011, pp. 1–29.
- [25] H. Sundmaeker, P. Guillemin, P. Friess, S. Woelffle, "چشم انداز و چالش های موجود بر سر راه تحقق بخشیدن به اینترنت اشیا"، کمیسیون اروپا- جامعه اطلاعاتی و رسانه ها DG، لوکزامبورگ، آلمان، Tech. Rep., Mar. 2010, (ISBN 978-92-79-15088-3, doi:10.2759/26127).

- [26] D. Georgakopoulos, P. Christen, A. Zaslavsky, C. Perera "CA4IOT: اطلاع رسانی مفهومی برای اینترنت اشیا" در *Proc. IEEE Int. Conf.*، شی اینترنتی، Nov. 2012, pp. 775–782.
- [27] سازمان تحقیقات علمی و صنعتی مشترک المنافع (CSIRO)، (2011) پدیده: شبکه حسگر توزیعی برای پدیده های پشتیبانی شده توسط مرکز پدیده شناسی گیاهی با وضوح بالا، مرکز CSIRO ICT، سنسورهای Csiro و شبکه های سنسور TCP، استرالیا، [Online]. در دسترس در <http://phenonet.com>.
- [28] D. Georgakopoulos, P. Christen, A. Zaslavsky, C. Perera "دریافت به عنوان یک مدل خدماتی برای شهرهای هوشمند پشتیبانی شده توسط اینترنت اشیا،" در *Proc. Trans. ETT*, 2013, pp. 1–13, doi: 10.1002/ett.2704
- [29] A. Sheth, C. Henson, H. Neuhaus, L. Lefort, M. Compton "یک نظرسنجی از خصوصیات معنایی سنسورها" در *Proc. 2nd Int.* شبکه سنسور معنایی کارگاه. *8th Int.* کنفرانس وب مفهومی. Oct. 2009, pp. 1–16
- [30] A. Schultz و C. Bizer "معیار SPARQL برلین"، *Int. J.* وب معنایی، *Inf. Syst.*, vol. 5, no. 2, pp. 1–24, 2009
- [31] S. Ran مدلی برای کشف خدمات وب با QoS، "مبادلات SIGecom"، vol. 4, no. 1, pp. 1–10, Mar. 2003
- [32] M. Sloman و D. Chalmers "ارزیابی کیفیت خدمات در محیط های محاسباتی تلفن همراه *IEEE Commun.* شگفتی های ارزیابی، vol. 2, no. 2, pp. 2–10, Apr. 1999.
- [33] D. Georgakopoulos, P. Christen, A. Zaslavsky, P. P. Jayaraman, C. Perera "Mosden: اینترنت اشیا میان افزار برای دستگاه های تلفن همراه با منابع محدود"، در *Proc. 47th HICSS* Jan. 2014, pp. 1–10

[34] P. Steggles و G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith به حرکت به

سوی درک بهتر از مفهوم و مفهوم شناسی " در *Proc. 1st Int. Symp.* محاسبات موجود دستی، 1999, pp.

304–307

[35] بنیاد Apache (2011). ریاضی رایج : کتابخانه ریاضیات رایج Apache, Forest Hill, MD, USA

[Online]. در دسترس <http://commons.apache.org/math/>:

[36] بنیاد نرم افزار Apache (نوامبر 2010). ApacheJena, Forest Hill, MD, USA [Online]. موجود

در <http://jena.apache.org/>:

[37] دولت استرالیا، اداره هواشناسی، (2012) داده های مرتبط محیطی تجربی منتشر شده توسط اداره هواشناسی،

ملبورن، استرالیا [آنلاین]. در دسترس <http://lab.environment.data.gov.au/>.

[38] D. Georgakopoulos, A. Salehi, P. Christen, A. Zaslavsky, C. Perera ثبت اطلاعات

سنسور از تلفن های همراه با استفاده از شبکه میان افزاری حسگر جهانی در *Proc. IEEE 23rd Int. Symp.*

, PIMRC, Sep. 2012, pp. 24–29.

[39] D. Georgakopoulos و P. Christen, M. Compton, A. Zaslavsky, C. Perera پیکربندی

معنایی مبتنی بر اینترنت میان افزاری اشیا " *Proc. 9th Int. Conf. SKG*, Oct. 2013, pp. 1–8.

[40] D. Georgakopoulos, A. Salehi, P. Christen, A. Zaslavsky, C. Perera اتصال تجهیزات

موبایل به میان افزارهای شبکه جهانی سنسور با استفاده از بسته بندی های تولید شده سیستم محور، " در *Proc.*

11th ACM Int Int. Eng دسترسی موبایلی بی سیم، May 2012, pp. 23–30.

کریت پررا (12S') مدرک کارشناسی (لیسانس) در رشته علوم کامپیوتر را از دانشگاه استافوردشایر، استوک-آن-ترنت، انگلستان در سال 2009 و مدرک **M.B.A.** را در رشته مدیریت بازرگانی از دانشگاه ویلز، کاردیف، بریتانیا، 2012 دریافت کرده است. او در حال حاضر مشغول ادامه دکتری در رشته علوم کامپیوتر در دانشگاه ملی استرالیا، کانبرا، استرالیا است. او با آزمایشگاه مهندسی اطلاعات، مرکز **CSIRO ICT** همکاری دارد. علایق پژوهشی او در حال حاضر عبارتند از: اینترنت چیزهایی که شامل محاسبات فراگیر و همه جانبه با تمرکز بر شبکه های حسگر و زمینه محاسبات آگاه می باشد. او یکی از اعضای انجمن ماشین های محاسباتی است.

ارکادی زاسلاوسکی (92M') راهنمای علوم منطقه علم مدیریت داده های معنایی با اطلاعات آزمایشگاه مهندسی، مرکز فناوری اطلاعات و ارتباطات، و **CSIRO** است. او استاد کمکی دانشگاه ملی استرالیا، کانبرا، و استرالیا، استاد پژوهشی دانشگاه تنولووری لارنس، فیلادلفیا جنوبی، **MI**، ایالات متحده آمریکا، و استاد ادبی دانشگاه نیو ساوت ولز، **Kensington**، استرالیا است. او در حال حاضر درگیر تعدادی از پروژه های تحقیقاتی اروپا و ملی است. او مدرک **M.Sc** را در مقطع کارشناسی ارشد ریاضیات کاربردی و مدرک کارشناسی ارشد علوم رایانه را از دانشگاه ایالتی تفلیس، جورجیا، روسیه در سال 1976 و مدرک دکتری علوم رایانه را از موسسه علوم کنترل مسکو، **U.S.S.R** در فرهنگستان علوم، مسکو، روسیه، در سال 1987 دریافت کرده است. او بیش از 300 نشریه تحقیقات منتشر کرده است. او یکی از اعضای ارشد **ACM** و عضو **IEEE** کامپیوتر و جوامع ارتباطات است.

چی هارولد لیو (10M') یک پژوهشگر کارمند در تحقیقات **IBM**، پکن، چین است. او مدرک **Ph.D** را از کالج امپریال لندن، لندن، انگلستان. و مدرک **B.Eng** را از دانشگاه **Tsinghua**، پکن، چین. دریافت کرده است. علاقه پژوهشی فعلی او معطوف به اینترنت چیزها، تجزیه و تحلیل داده های بزرگ، محاسبات تلفن همراه و شبکه های حساس، حسگر و شبکه های مشبک می باشد. او جایزه علمی برجسته جوانان سال 2013، اولین جایزه نوآوری **IBM** در سال 2012 و اولین جایزه ثبت اختراع آی بی ام در سال 2011 دریافت کرد و **EEWeb.com** به عنوان مهندس برجسته توسط در سال 2011 با او مصاحبه کرد. اختراعات او به طور گسترده در کنفرانس های بزرگ منتشر شد و مجلات و اتحادیه اروپا / ایالات متحده / چین 10 اخطار او را ثبت کردند. او به عنوان رئیس کل کارگاه های بین المللی

با IEEE SECON در سال 2013، IEEE WCNC در سال 2012 و به عنوان ACM UbiComp در سال 2011 خدمت کرده است.

مایکل کامپتون یک دانشمند تحقیقاتی در آزمایشگاه مهندسی اطلاعات مرکز ICT CSIRO است. او از زمان پیوستن به مرکز ICT در سال 2006، همچنان در تیم امنیت اطلاعات و حفظ حریم خصوصی مربوط به موضوعات شبکه بهداشت و سلامت حسگر و در سرویس داده های پروژه شبکه های حسگر با موضوع شبکه های حسگر بوده است و در حال حاضر طبق یک چارچوب معنایی برای وب سایت سنسور هیدرولوژیکی با موضوع IWIS کار می کند. علایق پژوهشی فعلی او شامل منطق و مشخصه طبق مدل و مستدل در مورد سیستم ها و الگوریتم ها به عنوان بخشی از سیستم هایی است که توصیفات تکراری پردازش مورد نیاز را اجرا می کنند. او در حال حاضر با فن آوری های وب معنایی جهت امنیت، سنسورها و ادغام داده ها کار می کند. او مدرک B.I.T (Hons) را از دانشگاه ملی استرالیا، کانبرا، استرالیا، در سال 2000 و مدرک Ph.D. را از دانشگاه کمبریج، کمبریج، انگلستان در سال 2007 دریافت کرده است.

پیتر کریستن دانشیار دانشکده تحقیقات علوم کامپیوتر، دانشگاه ملی استرالیا، کانبرا، و استرالیا است. او مدرک دیپلم در مهندسی علوم رایانه را از زوریخ ETH، زوریخ، سوئیس در سال 1995 و مدرک Ph.D. علوم رایانه ای را از دانشگاه بازل، بازل، سوئیس، در سال 1999 دریافت کرده است. منافع پژوهشی فعلی او شامل استخراج داده ها و تطبیق داده ها (تصمیم گیری واحد) است. او به ویژه به توسعه الگوریتم های مقیاس پذیر و بدون وقفه برای تطبیق داده ها و حریم خصوصی و جنبه های اطمینان از تطبیق داده ها و داده کاوی علاقه مند است. او بیش از 80 مقاله را چاپ کرده است، از جمله اینکه در سال 2012 کتاب تطبیق داده ها (Springer) را چاپ کرده است، او تهیه کننده اصلی پیوند فیبری (پیوند بیومدیكال آزادانه قابل گسترش)، تمیز کردن داده هایی با منبه باز، عدم تکثیر و ثبت سیستم پیوند می باشد.

دیمتریوس جروکوپولوس (91M') یک مدیر تحقیقاتی در مرکز ICI CSIRO است، که رئیس آزمایشگاه مهندسی اطلاعات است که در کانبرا و سیدنی مستقر است. او استادیار دانشگاه ملی استرالیا، کانبرا، و استرالیا است. او

پیش از آمدن به CSIRO در اکتبر 2008، موقعیت پژوهش و مدیریت خود در چندین آزمایشگاه صنعتی در ایالات متحده از سال 2000 تا 2008 را برگزار کرده است، او دانشمند ارشد Bellcore، NJ، ایالات متحده آمریکا بود جایی که مراکز تحقیقاتی Telcordias در آستین، TX، ایالات متحده آمریکا، و پوزنان، لهستان را مشخص کرد. از سال 1997 تا 2000، او مدیر فنی سازمان فناوری اطلاعات میکرو الکترونیک و شرکت کامپیوتری (MCC) و معمار اصلی پروژه هماهنگی زیرساخت مدیریت همکاری MCC MCC بود. او جایزه تعالی (Verizon) GTE و دو جایزه مقاله برجسته جامع کامپیوتری IEEE را دریافت کرده است و نامزد دریافت جایزه اسمیتسونیان علوم رایانه در علوم شد. او بیش از 100 مقاله مجله و کنفرانس منتشر کرده است. او در این زمینه نامزد کرده است. او بیش از 100 مقاله مجله و کنفرانس منتشر کرده است.

این مقاله، از سری مقالات ترجمه شده رایگان سایت ترجمه فا میباشد که با فرمت PDF در اختیار شما عزیزان قرار گرفته است. در صورت تمایل میتوانید با کلیک بر روی دکمه های زیر از سایر مقالات نیز استفاده نمایید:

لیست مقالات ترجمه شده ✓

لیست مقالات ترجمه شده رایگان ✓

لیست جدیدترین مقالات انگلیسی ISI ✓

سایت ترجمه فا ؛ مرجع جدیدترین مقالات ترجمه شده از نشریات معتبر خارجی