

نظارت هوشمند در اینترنت اشیاء با استفاده از یادگیری عمیق برای تحلیل و پیش‌بینی بلادرنگ

سینا نصراللهی^۱، سید ابراهیم دشتی^۲

^۱دانشکده فنی-واحد شیراز-دانشگاه آزاد اسلامی

sina2560462567@gmail.com

^۲دانشکده برق و کامپیوتر، واحد جهرم، دانشگاه آزاد اسلامی (مسول مکاتبات)

Sayed.Dashty@gmail.Com

چکیده : با فراگیر شدن و افزایش روز افزون استفاده و کاربردهای اینترنت اشیا مخصوصاً در صنعت نیاز به نظارت هوشمند و بلادرنگ به شدت احساس می شود، در این پژوهش ترکیب اینترنت اشیا (IoT) و یادگیری عمیق با مدل‌های ترکیبی CNN و RNN، در نظارت بلادرنگ و تحلیل پیش‌بینانه پیشنهاد شده است. این روش سیستمی نوین ارائه می‌دهد که داده‌های حسگرهای IoT را با استفاده از CNN برای تحلیل ویژگی‌های مکانی و RNN برای پردازش توالی‌های زمانی ترکیب می‌کند. این سیستم ناهنجاری‌ها را شناسایی کرده و خرابی‌های احتمالی را پیش‌بینی می‌کند، و امکان مداخلات پیشگیرانه را برای کاهش هزینه‌های عملیاتی و افزایش کارایی فراهم می‌آورد. ارزیابی سیستم با مجموعه داده‌هایی شامل بیش از یک میلیون رکورد (پارامترهایی نظیر دما، رطوبت و فشار) انجام شد. نتایج نشان داد که دقت شناسایی دما به ۹۹.۲٪ افزایش یافته است. این دقت بالا نشان‌دهنده توانایی سیستم در پیش‌بینی دقیق خرابی‌های بحرانی است.

کلید واژه‌ها: اینترنت اشیا، یادگیری عمیق، شبکه عصبی پیچشی، شبکه عصبی بازگشتی، شناسایی بلادرنگ ناهنجاری

در حالی که شبکه‌های عصبی پیچشی (CNN) به‌طور گسترده برای استخراج ویژگی‌های مکانی از داده‌های تصویری و داده‌های حسگر استفاده می‌شوند، شبکه‌های عصبی بازگشتی (RNN) توانایی پردازش داده‌های سری‌زمانی و شناسایی الگوهای وابسته به زمان را دارند. در گذشته، استفاده از هر کدام از این مدل‌ها به‌طور جداگانه برای تحلیل داده‌های مختلف رایج بوده است. پژوهش‌های قبلی نیز معمولاً به‌طور مجزا از CNN و RNN در تحلیل داده‌های IoT استفاده کرده‌اند. به‌عنوان مثال، در برخی مطالعات از CNN برای تحلیل ویژگی‌های تصویری و از RNN برای پیش‌بینی وضعیت‌های آینده سیستم‌های IoT استفاده شده است [۱۱][۲]. اما این تحقیق رویکرد نوآورانه‌ای را پیشنهاد می‌دهد که در آن ترکیب CNN و RNN برای پردازش همزمان داده‌های مکانی و زمانی مورد استفاده قرار می‌گیرد. این ترکیب موجب بهبود دقت در پیش‌بینی و شناسایی ناهنجاری‌ها و خرابی‌ها در سیستم‌های IoT می‌شود. به‌ویژه، استفاده همزمان از CNN برای تحلیل ویژگی‌ها و RNN برای پردازش داده‌های سری‌زمانی، قادر است مزایای هر دو مدل را به‌طور همزمان در یک سیستم واحد به کار گیرد و ضعف‌های موجود در استفاده از هر مدل به‌طور جداگانه را برطرف کند. این نوآوری در تحقیق ما، نه تنها موجب بهینه‌سازی پردازش داده‌ها در سیستم‌های IoT می‌شود، بلکه دقت پیش‌بینی‌ها و شناسایی خرابی‌ها را افزایش می‌دهد.

رشد اینترنت اشیاء (IoT) منجر به افزایش چشمگیر در تولید و انتقال داده‌ها شده است. با توجه به مقیاس و تنوع این داده‌ها، تحقیقات بسیاری برای مواجهه با چالش‌های مربوط به مدیریت و بهینه‌سازی جریان اطلاعات صورت گرفته است. از نظر مسئله اصلی، برخی تحقیقات به چالش حجم بالای داده‌های تولیدشده توسط دستگاه‌های اینترنت اشیاء پرداخته‌اند. به‌عنوان مثال، در یکی از مطالعات اشاره شده که انتقال همه داده‌ها برای تحلیل متمرکز، همیشه عملی یا مطلوب نیست [3]. این نگرانی در پژوهش دیگری نیز مطرح شده است، جایی که هزینه‌های مربوط به انتقال مداوم حجم زیاد داده به فضای ابری از نظر پهنای باند و هزینه‌های مالی مشخص شد. برای حل این چالش‌ها، پیشنهادات مختلفی ارائه شده است [3]. یکی از محبوب‌ترین رویکردها، پردازش در لبه (Edge Processing) است، جایی که داده‌ها محلی و نزدیک به نقطه تولید پردازش می‌شوند، پیش از آنکه برای انتقال تصمیم‌گیری شود. این روش نه تنها بار داده‌ها را در شبکه کاهش می‌دهد، بلکه پاسخ‌های بلادرنگی فراهم می‌کند که برای کاربردهای حساس ضروری است. با این حال، پردازش در لبه نیز محدودیت‌هایی دارد. دستگاه‌های اینترنت اشیاء، به‌ویژه آن‌هایی که برای صرفه‌جویی در انرژی و هزینه طراحی شده‌اند، ممکن است قدرت محاسباتی لازم برای انجام تحلیل‌های پیچیده را نداشته باشند.

این چالش اساس پیشنهاد ما برای توسعه سیستم‌های تطبیقی مبتنی بر هوش مصنوعی (AI) است. برخی محققان امکان استفاده از مدل‌های یادگیری ماشین سبک‌وزن برای دستگاه‌های محدود از نظر منابع را بررسی کرده‌اند. برای مثال، یک الگوریتم یادگیری ماشین مخصوص دستگاه‌های محدود طراحی شده است. با این وجود، اگرچه این پژوهش‌ها قابلیت استفاده از هوش مصنوعی در اینترنت اشیاء را نشان داده‌اند، همچنان کمبود سیستم‌های تطبیقی که بتوانند در بلادرنگ یاد بگیرند و تکامل یابند، احساس می‌شود. کمبود در مطالعات موجود به عدم وجود سیستم‌هایی برمی‌گردد که بتوانند داده‌ها را فیلتر و دسته‌بندی کرده و به‌صورت بلادرنگ با الگوهای متغیر تطبیق یابند. تطبیق‌پذیری بسیار مهم است، زیرا محیطی که دستگاه‌های اینترنت اشیاء در آن فعالیت می‌کنند، پویا است و الگوهای مشاهده‌شده یک روز ممکن است روز بعد مرتبط نباشند. پیشنهاد ما این شکاف را پر می‌کند. در حالی که از دانش انباشته‌شده در زمینه پردازش در لبه و مدل‌های سبک‌وزن هوش مصنوعی استفاده می‌کنیم، تمرکز ما بر روی قابلیت تطبیق‌پذیری است.

مروری بر کارهای پیشین نشان می‌دهد که اگرچه سیستم‌های متعددی برای نظارت بلادرنگ و تحلیل پیش‌بینانه وجود دارند، تعداد کمی از آن‌ها به‌طور مؤثری قابلیت تطبیق‌پذیری و یادگیری مداوم را، همانند سیستم ما، ادغام کرده‌اند. به‌ویژه، سیستم‌های موجود بیشتر بر جمع‌آوری و ارائه داده‌ها متمرکز هستند و از قابلیت‌های تحلیل پیشرفته و تطبیق‌پذیری بلادرنگی که ما ارائه می‌دهیم، بهره‌مند نیستند.

سیستم ما با یادگیری از داده‌های دریافتی، تطبیق با تغییرات، و تصمیم‌گیری‌های آگاهانه در مورد اینکه چه داده‌هایی منتقل شوند و چه زمانی، یک راه‌حل جامع‌تر و قدرتمندتر برای چالش مدیریت حجم داده‌ها در دستگاه‌های اینترنت اشیا ارائه می‌دهد پیشنهاد ما در زمینه سیستم تطبیقی مبتنی بر هوش مصنوعی، یک پیشرفت قابل توجه است و راه‌حلی پویا و تکاملی برای چالش مدیریت داده در دستگاه‌های اینترنت اشیا ارائه می‌دهد در ادامه چند روش مشابه دیگر را ذکر شده است.

روش اول هدف ارائه یک چارچوب هوش مصنوعی قابل تفسیر (XAI) برای تشخیص ناهنجاری‌ها در سیستم‌های اینترنت اشیا (IoT) طراحی شده است. چارچوب پیشنهادی از روش‌های پیشرفته XAI استفاده می‌کند تا علاوه بر افزایش دقت در شناسایی ناهنجاری‌ها، شفافیت و اعتمادپذیری مدل‌های هوش مصنوعی را نیز ارتقا دهد. در این راستا، ابزارهایی مانند SHapley Additive (SHAP) (exPlanations) برای تحلیل اهمیت ویژگی‌ها و تأثیر آن‌ها بر پیش‌بینی مدل و LIME (Local Interpretable Model-Agnostic (LIME Explanations) به منظور تفسیر محلی پیش‌بینی‌های خاص مورد استفاده قرار گرفته‌اند. علاوه بر این، کتابخانه‌های پایتون از جمله Keras برای ساخت و آموزش مدل‌ها، Pandas برای پردازش داده‌ها و Scikit-learn برای محاسبات و ارزیابی‌های آماری استفاده شده‌اند.

مجموعه داده‌های به کاررفته در این مطالعه شامل داده‌های MEMS از حسگرهای مبتنی بر تولید هوشمند و داده‌های N-BaIoT مرتبط با حملات بات‌نت در سیستم‌های IoT است. اگرچه چارچوب پیشنهادی مزایای بسیاری دارد، اما با چالش‌هایی مانند پیچیدگی و هزینه‌های محاسباتی بالا، وابستگی به داده‌های آموزشی باکیفیت، محدودیت در مقیاس‌پذیری، و دشواری در استفاده به صورت زمان واقعی مواجه است. همچنین، تفسیر برخی نتایج این چارچوب در شرایط خاص می‌تواند پیچیده باشد.

از جمله مزایای این چارچوب می‌توان به افزایش دقت در تشخیص ناهنجاری‌ها، تطبیق‌پذیری با داده‌های واقعی، تحلیل اهمیت ویژگی‌ها و افزایش شفافیت و اعتمادپذیری مدل‌ها اشاره کرد. این سیستم با استفاده از پارامترهای عملکردی نظیر Precision، Recall، و F1-score ارزیابی شده و توانایی بهبود عملکرد مدل‌های هوش مصنوعی در شناسایی ناهنجاری‌ها و تفسیر قابل اعتماد آن‌ها را نشان داده است.

نوآوری اصلی این روش، ترکیب روش‌های XAI برای افزایش شفافیت و ارائه چارچوبی است که علاوه بر تشخیص ناهنجاری‌ها، تحلیل اهمیت ویژگی‌ها را به شکلی دقیق و کاربردی امکان‌پذیر می‌سازد. تمرکز این مطالعه بر بهبود اعتمادپذیری و کارایی سیستم‌های IoT از طریق استفاده از هوش مصنوعی قابل تفسیر است [4].

روش دوم بررسی و پیاده‌سازی روش‌های تحلیل پیش‌بینانه برای سیستم‌های تولید هوشمند از طریق پلتفرمی به نام SERENA است. این پلتفرم که برای کاربردهای اینترنت اشیا صنعتی (IIoT) توسعه یافته است، از ابزاری به نام "RobotBox" برای شبیه‌سازی در محیط‌های رباتیک استفاده می‌کند. داده‌های مورد استفاده در این تحقیق عمدتاً شامل اطلاعات جمع‌آوری‌شده از حسگرهای صنعتی در کارخانه‌های تولیدی و سیستم‌های مبتنی بر IoT صنعتی هستند.

با وجود مزایای متعدد، این سیستم با چالش‌هایی همچون پیچیدگی پیاده‌سازی، نیاز به منابع محاسباتی بالا، وابستگی به داده‌های کیفی و متنوع، مدیریت پیچیده داده‌ها و تنظیمات اولیه، محدودیت در پیش‌بینی کامل خرابی‌ها، و چالش‌های مرتبط با امنیت و حریم خصوصی داده‌ها مواجه است. با این حال، مزایای استفاده از این پلتفرم شامل افزایش قابلیت پیش‌بینی و کاهش خرابی‌های غیرمنتظره، انعطاف و تطبیق‌پذیری با سیستم‌های مختلف، امکان تجزیه و تحلیل داده‌های حجیم و لحظه‌ای، قابلیت تجمیع داده‌های چنددوره‌ای، و پشتیبانی از تجسم داده و مدیریت ساده‌تر تعمیر و نگهداری است.

پلتفرم SERENA با بهره‌گیری از تحلیل دوره‌های زمانی متعدد، قادر است تغییرات تدریجی و الگوهای خرابی در تجهیزات را شناسایی کند. این سیستم از پارامترهای آماری نظیر میانگین، انحراف معیار، چولگی و کورتوزیس برای تجزیه و تحلیل داده‌ها استفاده می‌کند. همچنین، پارامترهایی مانند Aggregation، Horizon و Split size در فرآیند تحلیل داده‌های حسگرها به کار گرفته شده‌اند.

نوآوری این تحقیق در استفاده از پلتفرم SERENA و تحلیل پیش‌بینانه دوره‌های زمانی نهفته است که امکان شناسایی تغییرات تدریجی در سیستم‌های صنعتی و پیشگیری از خرابی‌های غیرمنتظره را فراهم می‌کند[5]. در این روش تأکید بر توانمندسازی تولید هوشمند از طریق تحلیل پیش‌بینانه و استفاده از پلتفرم‌های نوآورانه اینترنت اشیا صنعتی است.

روش سوم با هدف ارائه یک چارچوب جامع برای نگهداری پیش‌بینانه در سیستم‌های اینترنت اشیا صنعتی (IIoT) با بهره‌گیری از تکنیک‌های پیشرفته هوش مصنوعی طراحی شده است. این چارچوب بر اساس تجزیه و تحلیل داده‌های حسگرهای صنعتی توسعه یافته که اطلاعاتی همچون شرایط و عملکرد تجهیزات را ثبت و ارائه می‌کنند. ابزار شبیه‌سازی خاصی در این مقاله ذکر نشده است، اما تمرکز بر استفاده از داده‌های واقعی جمع‌آوری شده از حسگرهاست.

پیاده‌سازی این سیستم با چالش‌هایی نظیر پیچیدگی، نیاز به داده‌های حجیم و هزینه اولیه بالا همراه است. با این وجود، استفاده از این چارچوب مزایای قابل‌توجهی از جمله کاهش توقف‌های غیرمنتظره، افزایش طول عمر تجهیزات، صرفه‌جویی در هزینه‌ها و بهبود دقت تشخیص را به همراه دارد.

پارامترهای مورد بررسی در این سیستم شامل لرزش، دما، فشار، جریان مصرف برق و انتشار صوتی است که برای نظارت بر شرایط تجهیزات و پیش‌بینی خرابی‌ها استفاده می‌شود. برای تحلیل این داده‌ها، مقاله از شبکه‌های عصبی پیچشی (CNN) و شبکه‌های حافظه بلندمدت (LSTM) بهره می‌برد که به استخراج ویژگی‌های مهم و پیش‌بینی طول عمر باقیمانده تجهیزات کمک می‌کنند.

نوآوری این روش در ترکیب تکنیک‌های پیشرفته هوش مصنوعی مانند CNN و LSTM برای بهبود دقت پیش‌بینی و افزایش کارایی در نگهداری پیش‌بینانه تجهیزات صنعتی نهفته است[6]. این روش تأکید بر رویکرد نوین و کاربردی این تحقیق در بهبود نگهداری پیش‌بینانه در سیستم‌های IoT صنعتی دارد.

این روش با هدف ارائه یک مرور جامع بر راهکارهای مبتنی بر اینترنت اشیا (IIoT) و یادگیری ماشین (ML) در حوزه تولید هوشمند نگاشته شده است[7]. در این راستا از ابزار شبیه‌سازی دیجیتال توین (Digital Twin) برای مدل‌سازی سلول‌های تولیدی، شبیه‌سازی رفتار سیستم، پیش‌بینی خطاهای فرآیند و تنظیم متغیرهای کنترل به صورت تطبیقی استفاده شده است. مجموعه داده‌های مورد استفاده در این تحقیق شامل داده‌های مرتبط با ایمنی، نگهداری پیش‌بینی‌کننده، کنترل فرآیند و تولید افزودنی (چاپ سه‌بعدی) است.

از جمله معایب این روش‌ها می‌توان به پیچیدگی در پیاده‌سازی، چالش‌های مرتبط با مدیریت داده‌های بزرگ، مشکلات مربوط به کیفیت داده‌ها و نگرانی‌های امنیتی و حفظ حریم خصوصی اشاره کرد. با این حال، مزایایی همچون بهبود کارایی و کاهش هزینه‌ها، قابلیت مانیتورینگ بلادرنگ و انعطاف‌پذیری در کاربردهای مختلف، این رویکردها را به گزینه‌ای جذاب برای تولید هوشمند تبدیل کرده است.

پارامترهای کلیدی مورد بررسی در این تحقیق شامل شرایط فیزیکی اپراتورها مانند دما و رطوبت محیط کار است. هسته اصلی کار آن در استفاده ترکیبی از یادگیری ماشین و اینترنت اشیا (IIoT) در کاربردهای صنعتی است که می‌تواند نقش قابل‌توجهی در ارتقای تولید هوشمند و بهینه‌سازی فرآیندهای صنعتی ایفا کند.

این تحقیق بر تلفیق این فناوری‌ها و اهمیت آن‌ها در پیش رفت تولید هوشمند تأکید دارد.

محدودیت‌ها	مزایا	داده‌ها	روش پیشنهادی	مرجع / سال	ردیف
-محدودیت در مدل‌سازی وابستگی‌های زمانی بلندمدت	-دقت بالاتر از CNN و LSTM ترکیبی - معماری سبک‌تر از CNN- LSTM مناسب برای اجرا در زمان واقعی	Predictive Maintenance Dataset $\approx 10,000$ رکورد، داده حسگر موتور صنعتی	CNN-RNN برای تشخیص خطا و نگهداشت پیشگویانه	Sensors, 2024 (Predictive Maintenance of Robot Motors)	1
-هزینه محاسباتی بالا- دشواری استفاده در دستگاه‌های IoT با منابع محدود	-مدل‌سازی بهتر الگوهای فصلی و بلندمدت- بهبود معیارهای MAE, RMSE و R^2 نسبت به مدل‌های ساده	دیتاست موردی Microsoft سنسورهای صنعتی چندمتغیره	CNN-LSTM با temporal skip connection برای پیش‌بینی خرابی ماشین	Applied Sciences, 2022 (Industry 4.0 Case Study)	2

جدول ۱: دو تحقیق مشابه روش ترکیبی CNN/RNN

جدول ۱ دو پژوهش مشابه در زمینه‌ی ترکیب شبکه‌های CNN و RNN/LSTM در کاربردهای IoT و صنعت ۴.۰ ارائه می‌دهد. مطالعه نخست [8] از ترکیب CNN و RNN برای تشخیص خطا در ربات‌های صنعتی استفاده کرده است. مزیت اصلی آن سادگی معماری و سرعت بالای پردازش بلادرنگ است، اما محدودیت آن در شناسایی وابستگی‌های زمانی بلندمدت باقی مانده است. مطالعه [9] با بهره‌گیری از CNN-LSTM و افزودن temporal skip connection توانسته است الگوهای بلندمدت را با دقت بالاتری مدل کند، اما هزینه محاسباتی بالاتر و محدودیت در پیاده‌سازی روی دستگاه‌های IoT دارد.

پژوهش دوم، بر اجرای بهینه روی دستگاه‌های IoT با منابع محدود تمرکز شده است؛ بدین منظور تکنیک‌هایی نظیر Quantization و Pruning به کار گرفته شده تا مدل سبک‌تر و مناسب برای پردازش بلادرنگ گردد. بنابراین، کار حاضر تمایز خود را در حوزه‌ی کاربردی (اجرا روی دستگاه‌های IoT با محدودیت منابع) آشکار می‌سازد.

۲- مواد و روش ها



شکل ۱-فلوچارت مراحل پیش پردازش و اجرای الگوریتم CNN-RNN روی دستگاه های IoT

شکل ۱ فلوچارت مراحل اجرای الگوریتم ترکیبی CNN-RNN بر روی دستگاه های IoT را نشان می دهد. این فلوچارت شامل گام های اصلی از زمان دریافت داده خام تا تولید خروجی نهایی است. در مرحله نخست، داده ها از حسگرهای مختلف جمع آوری می شوند. سپس فرآیند پیش پردازش شامل حذف نویز، نرمال سازی و تقسیم داده ها صورت می گیرد تا داده ها آماده ورود به شبکه های عصبی شوند.

در گام بعد، ویژگی های مکانی با استفاده از شبکه CNN استخراج می شوند و ویژگی های زمانی از طریق شبکه RNN تحلیل می گردند. ادغام این دو دسته ویژگی (Feature Fusion) باعث می شود که مدل بتواند به طور هم زمان از اطلاعات مکانی و زمانی استفاده کند.

سپس، لایه تصمیم گیری (Fully Connected Layer) داده های ترکیبی را پردازش کرده و نتیجه نهایی شامل پیش بینی یا شناسایی ناهنجاری ارائه می شود. در نهایت، خروجی مدل می تواند به صورت فعال سازی یک عملکرد در دستگاه IoT یا ارسال نتیجه به سرور مرکزی مورد استفاده قرار گیرد.

همان طور که در فلوچارت مشاهده می شود، مراحل طراحی شده به گونه ای است که اجرای مدل بر روی دستگاه های IoT با منابع سخت افزاری محدود امکان پذیر باشد و در عین حال، سرعت و دقت لازم برای پردازش بلادرنگ نیز حفظ گردد.

آماده سازی داده ها یکی از مراحل کلیدی در توسعه سیستم های مبتنی بر یادگیری عمیق است که کیفیت و دقت مدل نهایی را تحت تأثیر قرار می دهد. در این پژوهش، داده های جمع آوری شده از حسگرهای IoT شامل داده های سری زمانی (مانند دما، رطوبت و فشار) و داده های تصویری از دوربین های IoT پردازش شده اند. مراحل پیش پردازش و آماده سازی داده ها به شرح زیر انجام شده است:

پیش پردازش داده های سری زمانی

نرمال سازی داده ها: داده های سری زمانی با استفاده از تکنیک Min-Max Normalization به بازه [۰, ۱] تبدیل شدند. این کار باعث می شود مقیاس داده های ورودی برای مدل یکنواخت شود و فرآیند آموزش مدل تسهیل گردد.

حذف نویز: از فیلترهای متحرک (Moving Average) برای حذف نویزهای تصادفی در داده ها استفاده شده است. این مرحله برای بهبود کیفیت داده های ورودی و جلوگیری از خطا در شناسایی ناهنجاری ها ضروری است.

تقسیم بندی داده ها: داده های سری زمانی به توالی هایی با طول ثابت (مانند ۵۰ یا ۱۰۰ زمان گام) تقسیم شدند تا بتوانند به صورت مؤثر توسط شبکه های RNN یا LSTM پردازش شوند.

برچسب‌گذاری داده‌ها: داده‌های ناهنجار با استفاده از قوانین مشخص یا مقادیر آستانه‌ای (Threshold) تعریف شده به عنوان "رخداد ناهنجاری" برچسب‌گذاری شدند.

پیش‌پردازش داده‌های تصویری

تغییر اندازه تصاویر: تمام تصاویر به ابعاد 224×224 پیکسل تغییر اندازه داده شدند تا با ورودی استاندارد شبکه‌های CNN سازگار باشند.

نرمال‌سازی رنگ: مقدار هر کانال رنگی R، G، B تصاویر به بازه $[0, 1]$ نرمال‌سازی شد تا مدل نسبت به تفاوت‌های نورپردازی مقاوم باشد.

افزایش داده: (Data Augmentation)

چرخش تصادفی: تصاویر در زوایای مختلف (تا $\pm 15^\circ$ درجه) چرخانده شدند.

برش تصادفی: (Random Cropping) بخش‌هایی از تصاویر به‌طور تصادفی انتخاب و بریده شدند.

تغییر شدت روشنایی: شدت نور تصاویر برای شبیه‌سازی شرایط واقعی تغییر یافت.

معکوس کردن افقی و عمودی: تصاویر به‌طور تصادفی برعکس شدند. این تکنیک‌ها باعث افزایش تنوع داده‌های آموزشی شده و از بیش‌برازش (Overfitting) جلوگیری می‌کند.

تشخیص و حذف مقادیر گمشده (Missing Values)

مقادیر گمشده در داده‌های سری‌زمانی با استفاده از روش‌های میانگین‌گیری یا برون‌یابی خطی (Linear Interpolation) جایگزین شدند. در مواردی که مقادیر زیادی از داده‌ها گمشده بودند، نمونه‌ها حذف شدند.

استخراج ویژگی‌ها (Feature Engineering)

ایجاد ویژگی‌های جدید: علاوه بر ویژگی‌های اصلی مانند دما، رطوبت و فشار، ویژگی‌های مشتقی مانند تغییرات نرخ (Rate of Change) و انحراف معیار محلی (Local Standard Deviation) برای بهبود قدرت مدل در شناسایی ناهنجاری‌ها استخراج شدند.

تحلیل همبستگی: (Correlation Analysis) ویژگی‌هایی که همبستگی بالایی با سایر ویژگی‌ها داشتند، حذف شدند تا از افزونگی داده‌ها کاسته شود.

تقسیم مجموعه داده‌ها

داده‌ها به سه بخش اصلی تقسیم شدند:

مجموعه آموزشی 70% (Training Set): داده‌ها برای آموزش مدل استفاده شدند.

مجموعه اعتبارسنجی 15% (Validation Set): داده‌ها برای تنظیم ابرپارامترها و جلوگیری از بیش‌برازش به کار رفتند.

مجموعه آزمایشی 15% (Test Set): داده‌ها برای ارزیابی عملکرد نهایی مدل مورد استفاده قرار گرفتند.

ذخیره و آماده‌سازی دسته‌های داده (Batch Preparation)

داده‌های پردازش‌شده به دسته‌های کوچک‌تر (Batches) تقسیم شدند تا به صورت موازی توسط GPU پردازش شوند. اندازه دسته‌ها (Batch Size) به صورت تجربی برابر با ۳۲ تعیین شد تا بین کارایی و استفاده از حافظه تعادل برقرار شود.

این مراحل آماده‌سازی داده‌ها، کیفیت داده‌های ورودی مدل را بهبود داده و سیستم را قادر می‌سازد تا به صورت بهینه و دقیق، ناهنجاری‌ها را در زمان واقعی شناسایی و پیش‌بینی کند.

فرمول CNN (۱)

$$Y_{i,j} = m, n \sum X_i + m, j + n \cdot W_{m,n} + b$$

که در آن:

$Y_{i,j}$: خروجی کانولوشن در موقعیت i, j است.

X : ورودی (تصویر یا داده) است.

W : فیلتر کانولوشن (کرnel) است.

b : بایاس است.

فرمول RNN (۲)

$$h_t = \tanh(W_h x_t + W_h h_{t-1} + b_h)$$

h_t : خروجی در زمان t است.

x_t : ورودی در زمان t است.

W_h : وزن‌های لایه‌ها هستند.

$W_h h$: وزن‌های بین حالات گذشته و حال هستند.

b_h : بایاس است.

۱-۲- تعریف مسئله

این سیستم به منظور نظارت بلادرنگ بر ماشین آلات و شرایط محیطی در محیط‌های صنعتی طراحی شده است. با رشد صنعت ۴.۰، [11] کارایی، ایمنی، و بهینه‌سازی فرآیندهای تولید به عوامل حیاتی تبدیل شده‌اند. بنابراین، وجود یک سیستم قوی و قابل اعتماد که بتواند اطلاعات دقیق و به‌موقعی از وضعیت ماشین آلات و شرایط محیطی ارائه دهد، بسیار ضروری است.

برای این منظور، چندین حسگر استفاده شده‌اند:

حسگر دما: نظارت بر دمای محیط و ماشین آلات را انجام می‌دهد تا از گرم شدن بیش از حد ماشین آلات جلوگیری کند.

حسگر رطوبت: برای نظارت بر سطح رطوبت محیط‌های صنعتی استفاده می‌شود، که از خوردگی ماشین آلات و آسیب به محصولات حساس به رطوبت جلوگیری می‌کند.

حسگر فشار: برای نظارت بر ماشین‌آلاتی که در شرایط فشار خاصی کار می‌کنند (مانند دیگ‌های بخار یا سیستم‌های هیدرولیک).

شتاب‌سنج سه‌محوره: ارتعاشات غیرعادی ماشین‌آلات را تشخیص می‌دهد که ممکن است نشانه‌ای از سایش، نقص یا نیاز به تعمیرات پیشگیرانه باشد.

دوربین اینترنت اشیا: نظارت تصویری از خط تولید ارائه می‌دهد تا موانع، خرابی‌ها، یا مشکلات دیگر به‌صورت بلادرنگ شناسایی شوند.

این حسگرها به‌طور استراتژیک در ماشین‌آلات حیاتی و نقاط خاص محیط تولید قرار داده شده‌اند. داده‌های جمع‌آوری شده توسط این حسگرها به سرور مرکزی منتقل می‌شود، جایی که مدل هوش مصنوعی آن‌ها را در بلادرنگ پردازش و تحلیل می‌کند. در صورت شناسایی هرگونه ناهنجاری، هشدارهایی برای کارکنان کارخانه ارسال می‌شود تا اقدامات اصلاحی انجام دهند.

۲-۲- مجموعه داده‌ها

برای توسعه، پیاده‌سازی، و اعتبارسنجی سیستم پیشنهادی، دو منبع اصلی برای جمع‌آوری داده‌ها استفاده شده است محیط‌های کنترل شده برای شبیه‌سازی شرایط خاص و محیط‌های تولید صنعتی برای جمع‌آوری داده‌های واقعی. این مجموعه داده‌ها به‌طور خاص طراحی شده‌اند تا کارایی و دقت مدل ترکیبی CNN-RNN را تحت شرایط مختلف ارزیابی کنند.

محیط‌های کنترل شده

در این محیط‌ها، داده‌ها با استفاده از الگوریتم‌های شبیه‌سازی تولید شدند تا شرایط عملیاتی دستگاه‌های IoT را بازسازی کنند.

تولید داده: پارامترهایی نظیر دما، رطوبت، فشار، و ارتعاش با استفاده از الگوریتم‌های شبیه‌سازی تولید شدند که تغییرات محیطی و شرایط استرس‌زا را مدل‌سازی می‌کنند.

محدوده داده‌ها: داده‌های شبیه‌سازی‌شده برای یک دوره سه‌ماهه شامل ۲۵,۰۰۰ رکورد جمع‌آوری شدند.

افزودن ناهنجاری‌ها: حدود ۵٪ ناهنجاری به‌طور عمدی به داده‌ها اضافه شد. این ناهنجاری‌ها شامل مقادیر افراطی یا داده‌های خارج از محدوده طبیعی بودند تا توانایی سیستم در شناسایی ناهنجاری‌ها آزمایش شود.

محیط‌های تولید صنعتی

داده‌های واقعی از حسگرهای IoT جمع‌آوری شدند که در کارخانه‌ای صنعتی نصب شده‌اند. این داده‌ها نمونه‌ای جامع و عملی برای تحلیل کارایی سیستم تحت شرایط واقعی فراهم می‌کنند.

منابع داده: حسگرهایی شامل حسگر دما، رطوبت، فشار، شتاب‌سنج و دوربین IoT.

مدت زمان جمع‌آوری: داده‌ها طی ۱۲ ماه با نرخ نمونه‌برداری منظم جمع‌آوری شدند.

ویژگی‌ها و حجم داده‌ها:

دما: ۴۰- تا ۸۵ درجه سانتی‌گراد با دقت ± 0.5 درجه.

رطوبت: ۰٪ تا ۱۰۰٪ با دقت $\pm 2\%$.

فشار: ۳۰۰ تا ۱۱۰۰ هکتوپاسکال با دقت ± 1 هکتوپاسکال.

شتاب‌سنج: داده‌های سه‌محوره برای پایش ارتعاشات.

دوربین: IoT تصاویر با کیفیت ۱۰۸۰p برای تحلیل‌های مکانی.

تعداد رکوردها: بیش از ۱ میلیون رکورد.

ناهنجاری‌های طبیعی: حدود ۰.۵٪ داده‌ها به دلیل اختلالات محیطی و خرابی حسگرها ناهنجاری دارند.

آماده‌سازی و مدیریت مجموعه داده‌ها

نرمال‌سازی و حذف نویز:

مقادیر دما، رطوبت، و فشار با استفاده از تکنیک Min-Max Normalization به بازه [۰, ۱] مقیاس‌بندی شدند.

داده‌های تصویری از نظر روشنایی و کنتراست نرمال‌سازی شدند.

افزایش داده‌ها (Data Augmentation)

برای تصاویر: چرخش تصادفی، تغییر شدت روشنایی، و برش تصادفی برای افزایش تنوع.

برای سری‌زمانی: شبیه‌سازی نویز و داده‌های اضافی برای آزمایش استحکام سیستم.

برچسب‌گذاری (Labeling)

داده‌های دارای ناهنجاری به‌عنوان داده‌های "رخداد غیرعادی" برچسب‌گذاری شدند.

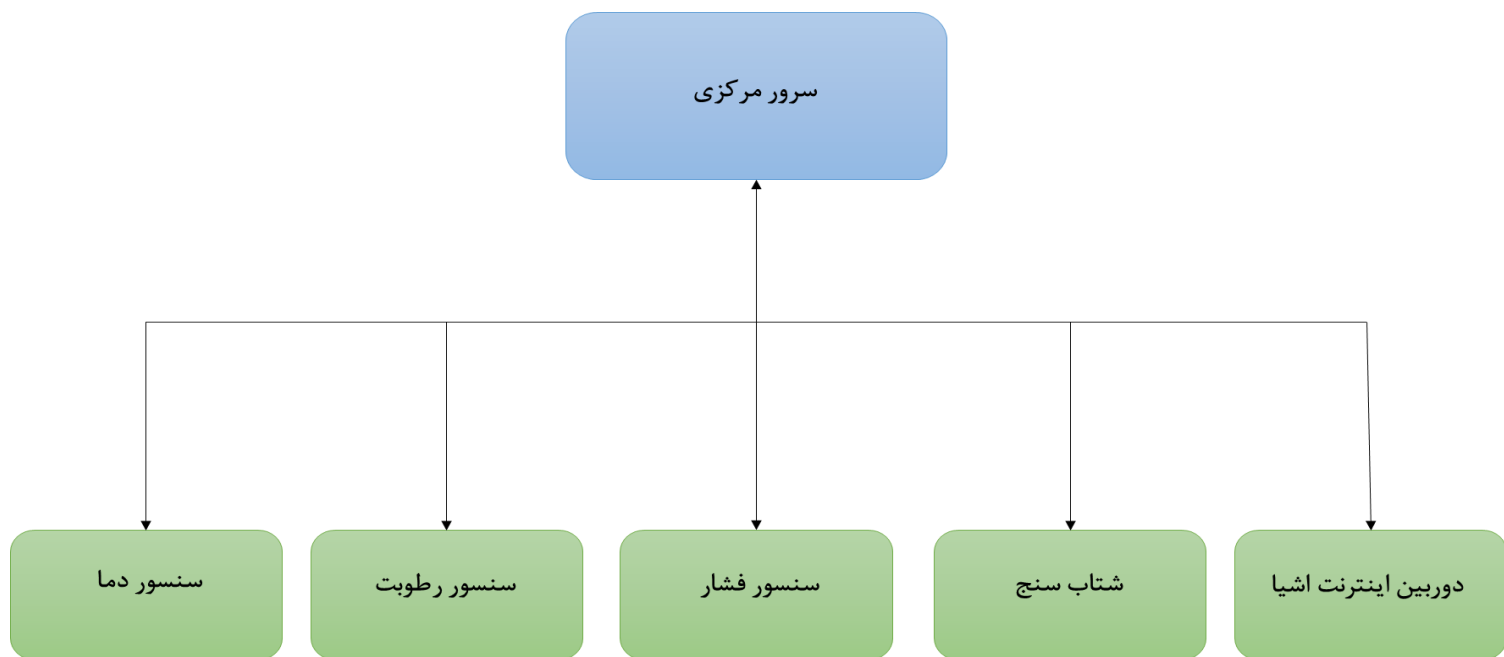
تصاویر با مشکلات مشخص (مانند انسداد دید) به دسته ناهنجاری اضافه شدند.

طراحی ناهنجاری‌ها برای آزمایش

ناهنجاری‌های سری‌زمانی: ورود مقادیر خارج از محدوده یا تغییرات ناگهانی در نرخ افزایش/کاهش.

ناهنجاری‌های تصویری: تصاویر با انسداد جزئی، نور کم، یا تغییرات شدید روشنایی.

این طراحی جامع مجموعه داده‌ها تضمین می‌کند که سیستم می‌تواند به‌طور مؤثر ناهنجاری‌ها را شناسایی کرده و در شرایط عملیاتی متنوع سازگاری و دقت خود را حفظ کند.



شکل ۲: معماری کلی محیط اجرا

۳- روش

پیش پردازش داده‌ها: پیش پردازش داده‌ها یک مرحله حیاتی در آماده سازی داده‌ها برای تحلیل و مدل سازی است که شامل تکنیک‌ها و ابزارهای مختلف برای پاک سازی، تبدیل و آماده سازی مجموعه داده‌ها می شود. هدف اصلی این فرآیند، بهبود کیفیت داده‌ها و تسهیل استفاده بهینه از آن‌ها در مدل های یادگیری عمیق ترکیبی (CNN-RNN) است. در این پژوهش، مجموعه داده‌ها از حسگرهای IoT و داده‌های تصویری تحت پیش پردازش دقیق قرار گرفتند تا برای آموزش و ارزیابی سیستم آماده شوند.

تشخیص و حذف داده‌های پرت

روش IQR بین چارکی: برای شناسایی و حذف داده‌های پرت، از روش قاعده محدوده بین چارکی (Interquartile Range) استفاده شد. این روش داده‌هایی که به طور قابل توجهی از الگوهای کلی داده‌ها انحراف دارند را شناسایی کرده و حذف می کند تا مدل با داده‌های نادرست آموزش نبیند.

مدیریت داده‌های گمشده

روش Imputation برای پر کردن مقادیر گمشده در داده‌ها از روش‌های مختلفی مانند میانگین، میانه یا مد استفاده شد. این روش بسته به نوع داده و توزیع آن در ستون‌های خاص انتخاب شد تا کمترین تأثیر را بر دقت مدل داشته باشد.

ابزارهای مورد استفاده

Pandas برای مدیریت داده‌ها، پاک‌سازی و انجام تحلیل‌های آماری ابتدایی [10][11].

Scikit-learn برای نرمال‌سازی داده‌ها و انجام تکنیک‌های پیش‌پردازش پیشرفته مانند استانداردسازی و رمزگذاری One-Hot.

تبدیلات یا نرمال‌سازی‌های اعمال شده

استانداردسازی: برای داده‌های عددی، از استانداردسازی استفاده شد تا ویژگی‌ها میانگین صفر و انحراف معیار یک داشته باشند. این کار برای اطمینان از همگرایی بهتر الگوریتم‌های یادگیری عمیق [12] و مقایسه مؤثر ویژگی‌ها در مدل‌های ترکیبی CNN-RNN ضروری است.

نرمال‌سازی Min-Max: برای ویژگی‌های عددی که دامنه‌های بسیار متفاوتی دارند، از نرمال‌سازی Min-Max برای تبدیل مقادیر به بازه [0, 1] استفاده شد. این تکنیک کمک می‌کند تا رابطه بین مقادیر حفظ شود و داده‌ها به‌طور یکنواخت مقیاس‌بندی شوند.

رمزگذاری One-Hot

ویژگی‌های دسته‌ای، مانند نوع دستگاه یا شرایط محیطی، با استفاده از روش One-Hot Encoding به بردارهای باینری تبدیل شدند. این تکنیک برای پردازش داده‌های غیرعددی که توسط مدل‌های یادگیری عمیق قابل استفاده نیستند، ضروری است.

آماده‌سازی داده‌ها برای مدل‌های ترکیبی CNN-RNN

داده‌های تصویری: تصاویر جمع‌آوری شده از دوربین‌های IoT ابتدا با استفاده از نرمال‌سازی رنگ و تغییر اندازه به ابعاد ثابت 224×224 پیکسل پردازش شدند.

داده‌های سری‌زمانی: داده‌های حسگرهای IoT به توالی‌های زمانی تقسیم شدند تا توسط RNN و LSTM برای شناسایی الگوهای زمانی پردازش شوند.

این فرآیند پیش‌پردازش، داده‌ها را به‌طور مؤثر برای آموزش و ارزیابی مدل‌های ترکیبی آماده می‌کند و تضمین می‌کند که مدل‌های یادگیری عمیق بتوانند به‌طور دقیق و کارآمد به شناسایی ناهنجاری‌ها و پیش‌بینی‌های بلندمدت بپردازند.

ابزار/روش	توضیحات	تکنیک پیش‌پردازش
قانون IQR	شناسایی و حذف مقادیری که از الگو و حالت عادی انحراف دارند	تشخیص داده پرت
Pandas میانگین، میانه، مد	پر کردن مقادیر گمشده با آمار توصیفی مانند میانگین، میانه و مد	جای‌گذاری مقادیر گمشده
Scikit-learn(Standard-Scaler)	بدیل داده به حالتی که میانگین برابر با ۰ و انحراف معیار برابر با ۱ باشد	استانداردسازی
Scikit-learn(MinMaxScaler)	مقیاس‌گذاری ویژگی‌ها به بازه [۰, ۱]	نرمال‌سازی مین-مکس
Pandas(get_dummies)	تبدیل ویژگی‌های دسته‌ای به قالب باینری	رمزگذاری وان-هات

جدول ۲: خلاصه روش‌های پیش‌پردازش استفاده شده

۳-۱- روش‌شناسی و طراحی هوش مصنوعی برای سیستم فیلترینگ

استفاده از هوش مصنوعی در سیستم‌های اینترنت اشیا (IoT) روش‌های مدیریت و پردازش داده‌ها را به‌طور قابل‌توجهی متحول کرده است. در این پژوهش، برای مدیریت چالش‌های مربوط به پردازش داده‌های بلندمدت و بهینه‌سازی استفاده از پهنای باند، یک سیستم فیلترینگ مبتنی بر هوش مصنوعی طراحی شده است. این سیستم ترکیبی از شبکه‌های عصبی پیچشی (CNN) و شبکه‌های عصبی بازگشتی (RNN) را برای تحلیل داده‌های تصویری و سری‌زمانی به‌صورت هم‌زمان به کار می‌برد. انتخاب صحیح مدل، تنظیم پارامترها و روند آموزش تأثیر مستقیمی بر کارایی و دقت سیستم دارد.

هسته اصلی سیستم: ترکیب CNN و RNN

برای پردازش داده‌های تصویری و سری‌زمانی، از ترکیب دو مدل CNN و RNN استفاده شده است:

شبکه‌های عصبی پیچشی (CNN) برای استخراج ویژگی‌های مکانی از داده‌های تصویربرداری‌های IoT استفاده می‌شود. این مدل به شناسایی الگوهای پیچیده و ویژگی‌های بصری در تصاویر کمک می‌کند.

شبکه‌های عصبی بازگشتی (RNN) برای پردازش داده‌های سری‌زمانی و شناسایی الگوهای زمانی در داده‌های حسگرها طراحی شده است. این مدل توانایی حفظ "حافظه" از داده‌های قبلی در توالی‌ها را دارد که برای شناسایی ناهنجاری‌ها و پیش‌بینی‌های بلندمدت حیاتی است.

پارامترها و معماری مدل

معماری: RNN

لایه ورودی: داده‌های حسگرها از هر دستگاه IoT دریافت می‌شوند و به لایه‌های بعدی پردازش می‌شوند. اندازه این لایه بستگی به تعداد ویژگی‌های ورودی دارد.

لایه‌های مخفی: مدل از سه لایه مخفی استفاده می‌کند که هر کدام شامل ۱۲۸، ۶۴ و ۳۲ نورون به ترتیب هستند. هر لایه از تابع فعال‌سازی Rectified Linear Unit (ReLU) برای کاهش مشکل محوشدگی گرادینان و افزایش کارایی استفاده می‌کند.

لایه خروجی: این لایه پیش‌بینی نهایی سیستم را ارائه می‌دهد. از تابع فعال‌سازی سیگموئید برای ارائه احتمال ناهنجاری استفاده می‌شود.

روند آموزش مدل

آموزش مدل RNN به صورت تکراری انجام می‌شود، جایی که وزن‌ها و بایاس‌ها برای حداقل کردن تفاوت پیش‌بینی‌ها و مقادیر واقعی به روزرسانی می‌شوند:

بهینه‌ساز: از بهینه‌ساز Adam برای سرعت بخشیدن به همگرایی و دستیابی به عملکرد پایدار استفاده می‌شود. نرخ یادگیری اولیه برابر با ۰.۰۰۱ تنظیم شده است.

مدت آموزش: مدل برای ۵۰ دوره (epoch) آموزش داده می‌شود تا به همگرایی مناسب برسد.

تابع خطا و جلوگیری از بیش‌برازش

تابع خطا از Binary Cross Entropy برای ارزیابی تفاوت بین پیش‌بینی‌ها و مقادیر واقعی در مسائل طبقه‌بندی دودویی (شناسایی ناهنجاری) استفاده شده است.

جلوگیری از بیش‌برازش: برای کاهش خطر بیش‌برازش، به‌ویژه در مدل‌های RNN، از تکنیک Dropout استفاده می‌شود که به‌طور تصادفی نورون‌ها را غیرفعال می‌کند و به شبکه اجازه می‌دهد تا نمایه‌های مقاوم‌تر و کلی‌تری تولید کند.

معیارهای اصلی طراحی سیستم

سیستم فیلترینگ بر اساس دو معیار اصلی طراحی شده است:

۱. حذف داده‌های تکراری: داده‌هایی که تغییرات قابل توجهی ندارند یا تکراری هستند به‌عنوان داده‌های زائد شناخته شده و فیلتر می‌شوند.

۲. اولویت‌بندی داده‌ها: داده‌ها بر اساس اهمیتشان اولویت‌بندی می‌شوند. داده‌های حیاتی یا هشدارهای ناهنجاری اولویت بالا دارند، در حالی که داده‌های روتین برای انتقال کمتر مکرر گروه‌بندی می‌شوند.

تصمیم‌گیری بلادرنگ

سیستم پس از ارزیابی داده‌ها در طی چند میلی‌ثانیه تصمیم می‌گیرد که داده‌ها را منتقل، گروه‌بندی یا حذف کند. این تصمیم‌گیری بلادرنگ به دستگاه‌های IoT این امکان را می‌دهد که به شرایط متغیر پاسخ سریع و مناسبی بدهند. سیستم با استفاده از مدل ترکیبی CNN-RNN قادر است به‌طور مؤثر ناهنجاری‌ها را شناسایی کرده و در همان زمان از منابع شبکه بهینه استفاده کند.

این طراحی سیستم با استفاده از ترکیب هوش مصنوعی و مدل‌های یادگیری عمیق، راه‌حلی مناسب برای پردازش داده‌های بلادرنگ و شناسایی ناهنجاری‌ها در محیط‌های صنعتی فراهم می‌کند.

۲-۳ - عملیاتی سازی روش روی دستگاه

اجرای مدل‌های هوش مصنوعی روی دستگاه‌های IoT چالشی بزرگ به حساب می‌آید، زیرا این دستگاه‌ها محدودیت‌های منابع قابل توجهی دارند که نسبت به سرورها یا ورک‌استیشن‌های سنتی متفاوت است. هدف اصلی در این فرآیند، تضمین عملکرد بهینه مدل در زمان واقعی (بلادرنگ) بدون کاهش دقت یا عملکرد دستگاه است. در این پژوهش، از مدل‌های یادگیری عمیق ترکیبی CNN-RNN برای شناسایی ناهنجاری‌ها و پیش‌بینی رفتار دستگاه‌ها استفاده شده است. این مدل‌ها به‌طور ویژه برای اجرا بر روی دستگاه‌های IoT بهینه شده‌اند تا بتوانند داده‌ها را به‌طور مؤثر در زمان واقعی پردازش کنند.

مراحل یکپارچه‌سازی مدل:

پس از آموزش مدل CNN-RNN، فرآیند تبدیل مدل به قالب مناسب برای دستگاه‌های IoT با منابع محدود آغاز شد. برای این منظور از ابزارهایی مانند TensorFlow Lite استفاده گردید [13] که امکان تبدیل مدل‌های TensorFlow به نسخه‌های بهینه‌شده برای دستگاه‌های لبه (Edge Devices) را فراهم می‌آورد. سپس مدل تبدیل‌شده به فریم‌ور دستگاه IoT ادغام شده و با نوشتن کدی خاص، مدل در حافظه دستگاه بارگذاری می‌شود و داده‌های ورودی پردازش می‌گردند.

فرآیند ادغام مدل در دستگاه اینترنت اشیا با استفاده از روش ترکیبی یادگیری عمیق

ورودی داده (Input Data):

این بخش وظیفه دریافت داده‌های اولیه از دستگاه‌های IoT، مانند داده‌های حسگرها یا تصاویری است که توسط دوربین‌ها گرفته می‌شود. این داده‌ها ممکن است به‌صورت سری‌زمانی یا داده‌های مکانی (مانند تصاویر) باشند.

پیش‌پردازش داده (Data Preprocessing):

داده‌های ورودی در این مرحله تمیز و استانداردسازی می‌شوند. عملیات‌هایی مانند حذف نویز، نرمال‌سازی داده‌ها و تقسیم داده‌ها به بخش‌های مناسب برای ورود به CNN یا RNN در این مرحله انجام می‌شود.

ادغام ویژگی‌ها (Feature Fusion):

ویژگی‌های استخراج‌شده توسط CNN و RNN در این مرحله با هم ترکیب می‌شوند. این ترکیب به مدل اجازه می‌دهد از اطلاعات مکانی و زمانی به‌طور هم‌زمان استفاده کند و یک نمای کلی‌تر و جامع‌تر از داده‌ها به دست آورد.

لایه تصمیم‌گیری (Decision Layer):

در این بخش، داده‌های ترکیبی پردازش‌شده وارد یک لایه کاملاً متصل (Fully Connected Layer) می‌شوند که وظیفه نهایی‌سازی پیش‌بینی یا تصمیم‌گیری را دارد [14]. نتایج این لایه می‌تواند شامل دسته‌بندی، پیش‌بینی سری‌زمانی یا اجرای یک عملکرد خاص در دستگاه IoT باشد.

خروجی نهایی (Output):

این بخش نتیجه نهایی را در قالبی مانند پیش‌بینی دسته‌بندی، پیش‌بینی مقدار، یا فعال‌سازی عملکرد دستگاه ارائه می‌دهد.

مزایای روش ترکیبی CNN و RNN نسبت به DNN:

کاهش پیچیدگی محاسباتی:

برخلاف DNN، ترکیب CNN و RNN از ساختارهای تخصصی برای پردازش داده‌ها استفاده می‌کند که محاسبات کمتری را به همراه دارد و برای دستگاه‌های IoT با منابع محدود مناسب‌تر است.

بهره‌برداری از ویژگی‌های مکانی و زمانی:

CNN برای داده‌های مکانی (تصویری) و RNN برای داده‌های زمانی (سری زمانی یا وابسته به توالی) طراحی شده‌اند. ترکیب این دو شبکه امکان استفاده هم‌زمان از اطلاعات زمانی و مکانی را فراهم می‌کند، در حالی که DNN در شناسایی این ویژگی‌ها محدودتر عمل می‌کند.

کاهش اندازه مدل:

DNN به دلیل تعداد زیاد لایه‌ها و وزن‌ها معمولاً حجیم‌تر است. حذف DNN و استفاده از CNN و RNN باعث می‌شود که مدل نهایی سبک‌تر باشد و با حافظه کمتر (مانند SRAM) سازگار باشد.

عملکرد بهتر در زمان واقعی (Real-Time):

روش ترکیبی CNN و RNN با بهینه‌سازی داده‌ها و پردازش آن‌ها در دسته‌های مرتبط، عملکرد بهتری در پردازش داده‌ها به‌صورت بلادرنگ دارد. این موضوع برای دستگاه‌های IoT که نیازمند پاسخ سریع هستند، حیاتی است.

انعطاف‌پذیری بیشتر در طراحی مدل:

در این روش، می‌توان از CNN و RNN به‌صورت موازی یا ترتیبی استفاده کرد و آن‌ها را با معماری‌های مختلف ترکیب کرد تا دقت و کارایی مدل افزایش یابد. این انعطاف در DNN کمتر است.

صرفه‌جویی در مصرف انرژی:

دستگاه‌های IoT به دلیل محدودیت انرژی، نیازمند مدل‌های سبک‌تر و کم‌مصرف‌تر هستند. حذف DNN و استفاده از CNN و RNN که بهینه‌تر عمل می‌کنند، مصرف انرژی را کاهش می‌دهد.

بهینه‌سازی‌ها برای عملکرد بهتر:

با توجه به محدودیت‌های سخت‌افزاری دستگاه‌های IoT، چندین تکنیک به‌منظور تضمین عملکرد بلادرنگ به‌کار گرفته شد:

هرس کردن (Pruning): حذف وزن‌ها یا نورون‌هایی که تأثیر کمی بر دقت دارند. این تکنیک باعث کاهش اندازه مدل و افزایش سرعت استنتاج می‌شود.

کمیت‌سازی (Quantization): تبدیل وزن‌ها و بایاس‌های مدل از فرمت ۳۲ بیت به فرمت‌های کوچکتر مانند ۸ بیت. این تکنیک باعث کاهش حجم مدل و افزایش سرعت استنتاج به دلیل کاهش بار محاسباتی می‌شود.

کامپایل مدل در زمان اجرا (On-the-Fly Compilation): استفاده از کامپایلرهای تخصصی برای ترجمه مدل به دستورالعمل‌های مخصوص دستگاه و بهینه‌سازی عملکرد آن.

بهینه‌سازی حافظه: بارگذاری مدل در سریع‌ترین حافظه موجود روی دستگاه مانند SRAM به جای حافظه فلش یا خارجی برای تسریع در فرآیند پردازش.

پردازش دسته‌ای (Batch Processing): تا حد امکان داده‌ها به‌صورت دسته‌ای پردازش شدند تا از قابلیت‌های موازی‌سازی دستگاه بهره‌برداری شود و کارایی سیستم بهبود یابد.

این بهینه‌سازی‌ها اطمینان می‌دهند که مدل به‌طور مؤثر و در زمان واقعی روی دستگاه IoT اجرا شده و استنتاج‌ها به سرعت و دقت انجام می‌شوند.

۳-۳- اعتبار سنجی و آزمایش

برای اطمینان از صحت عملکرد سیستم، به‌ویژه در محیط‌های صنعتی با شرایط متغیر، آزمایش‌های مختلفی طراحی و اجرا شد:

آزمایش در محیط کنترل‌شده

در این سناریو، دستگاه IoT با مدل یکپارچه در محیطی آزمایشگاهی یا مشابه آن آزمایش می‌شود. در این شرایط، تمام متغیرها و شرایط به‌طور دقیق کنترل و نظارت می‌شوند. این نوع آزمایش امکان ارزیابی عملکرد سیستم در شرایط ایده‌آل را فراهم می‌کند. به‌عنوان مثال، می‌توان مقادیر خاص دما و رطوبت را شبیه‌سازی کرده و بررسی کرد که آیا دستگاه پاسخ‌های صحیح و مورد انتظار را ارائه می‌دهد یا خیر. این آزمایش برای شناسایی مشکلات بنیادی یا خرابی‌های آشکار کاربرد دارد.

آزمایش در محیط شبیه‌سازی‌شده واقعی

در این نوع آزمایش، دستگاه در محیطی شبیه‌سازی می‌شود که شرایط واقعی را برای سیستم فراهم می‌آورد، به‌طوری که دستگاه در شرایط نزدیک به واقعیت کار کند. این آزمایش برای ارزیابی عملکرد سیستم در مواجهه با شرایط پیچیده‌تر و پویاتر طراحی شده است. در این محیط، نظارت و کنترل همچنان وجود دارد، اما دستگاه باید توانایی تعامل با شرایط متغیر و غیرقابل پیش‌بینی را داشته باشد.

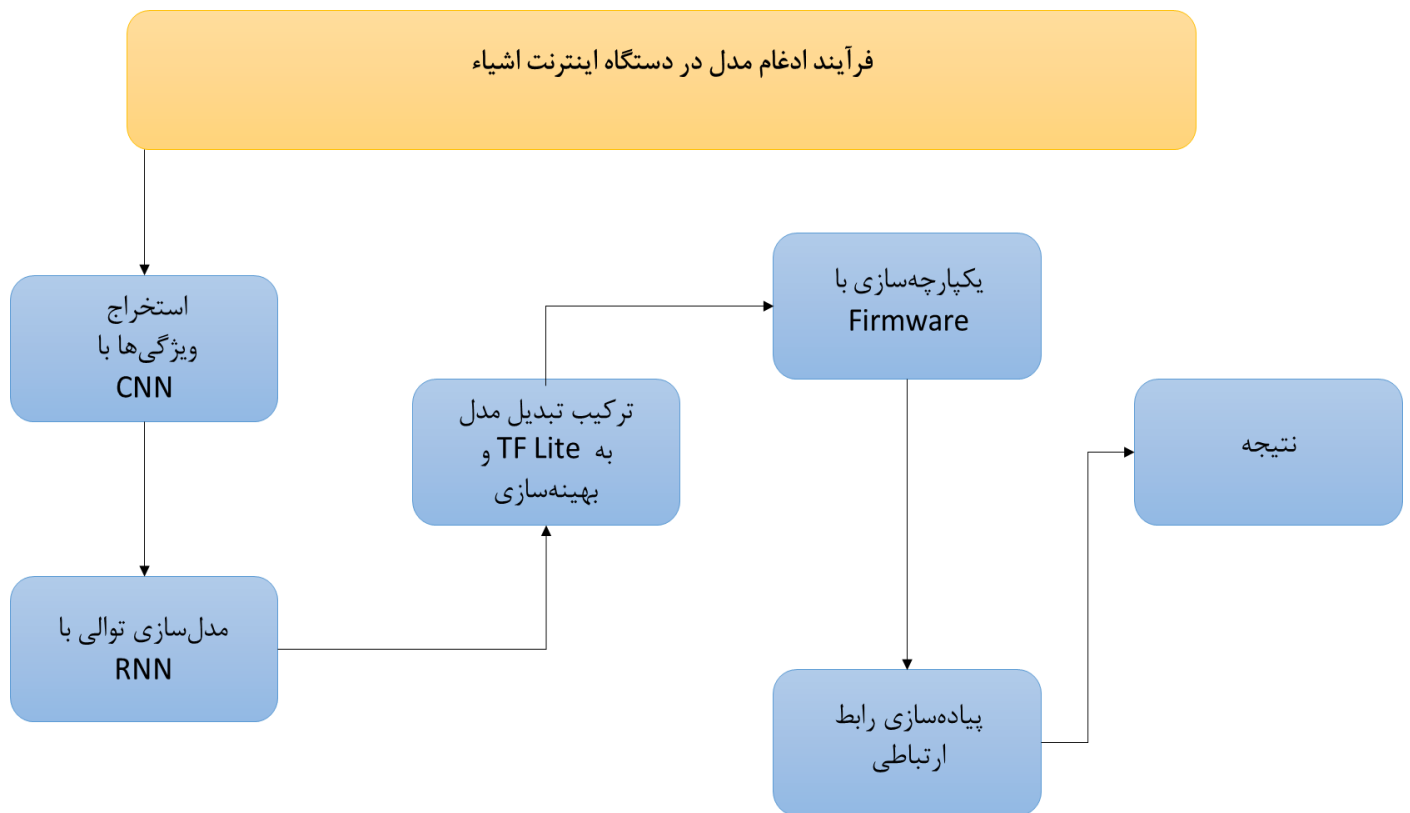
آزمایش در محیط واقعی

در نهایت، سیستم در محیط صنعتی واقعی مورد آزمایش قرار می‌گیرد تا عملکرد آن در شرایط عملیاتی واقعی ارزیابی شود. این مرحله برای تأیید صحت مدل در شرایط بدون شبیه‌سازی و بررسی اثربخشی آن در شناسایی ناهنجاری‌ها و پیش‌بینی مشکلات به‌کار می‌رود.

نتیجه‌گیری از آزمایش‌ها:

آزمایش‌ها در محیط‌های کنترل‌شده، شبیه‌سازی‌شده و واقعی نشان دادند که سیستم هوش مصنوعی با استفاده از مدل‌های ترکیبی-CNN و RNN توانایی شناسایی ناهنجاری‌ها را با دقت بالا در بلادرنگ دارد. این آزمایش‌ها همچنین کمک کردند تا مشکلات احتمالی سیستم شناسایی شوند و بهینه‌سازی‌های بیشتری انجام گردد تا سیستم در شرایط مختلف و در زمان واقعی به‌طور مؤثر عمل کند.

این فرآیند اعتبارسنجی و آزمایش کمک می‌کند تا اطمینان حاصل شود که سیستم به‌طور مؤثر و با دقت بالا در محیط‌های صنعتی مختلف اجرا می‌شود و به‌طور مؤثری ناهنجاری‌ها و مشکلات احتمالی را پیش‌بینی و مدیریت می‌کند.



شکل ۳: فرآیند ادغام مدل در دستگاه اینترنت اشیا

کنترل برخی متغیرها:

این نوع آزمایش، دیدگاه واقع‌بینانه‌تری از عملکرد سیستم ارائه می‌دهد و به شناسایی مشکلاتی که ممکن است در محیط کنترل شده آشکار نشوند، کمک می‌کند. به عنوان مثال، یک سناریو می‌تواند شامل استقرار دستگاه برای مدت کوتاهی در یک کارخانه یا خانه هوشمند باشد.

معیارهای ارزیابی عملکرد سیستم:

برای ارزیابی جامع عملکرد سیستم، از چندین معیار کلیدی استفاده می‌شود که اطلاعات ارزشمندی درباره جنبه‌های مختلف سیستم فراهم می‌کنند. این معیارها به ما کمک می‌کنند تا کارایی سیستم را در شرایط واقعی و عملیاتی سنجیده و بهینه‌سازی‌های لازم را اعمال کنیم:

Accuracy

این معیار نسبت پیش‌بینی‌های صحیح مدل به کل پیش‌بینی‌ها را اندازه‌گیری می‌کند به فرمول (3) دقت کنید.

$$Accuracy = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} \quad (3)$$

Latency

با توجه به اجرای سیستم در دستگاه‌های IoT با منابع محدود، زمان تأخیر (زمان پردازش یک ورودی و تولید خروجی) اهمیت زیادی دارد. این معیار بر حسب میلی‌ثانیه (ms) اندازه‌گیری می‌شود؛ هرچه کمتر باشد، سیستم کاراتر است [14].

Resource

Usage

اطلاع از میزان استفاده از منابع مانند CPU، حافظه و انرژی حیاتی است، به‌ویژه در دستگاه‌های دارای منابع محدود. این معیار می‌تواند به صورت درصد یا مقدار مطلق بیان شود.

Recall

نسبت موارد مثبت واقعی که به‌درستی شناسایی شده‌اند. این معیار برای کاربردهایی که پیامدهای خطای منفی (False Negative) زیاد است، اهمیت دارد. به فرمول (2) دقت کنید.

$$Recall = \frac{\text{True postivies}}{\text{True positive+False negatives}} \quad (4)$$

Precision

نسبت موارد مثبت شناسایی‌شده به کل موارد مثبت پیش‌بینی‌شده را اندازه‌گیری می‌کند [15]. به فرمول (3) دقت کنید.

$$Precision = \frac{\text{True postivies}}{\text{True positive+False postives}} \quad (5)$$

F1-Score

این معیار ترکیبی از دقت و یادآوری است و برای شرایطی که کلاس‌ها نامتعادل هستند، مفید است. به فرمول (4) دقت کنید.

$$F1 - Score = 2 * \frac{\text{Precision*Recall}}{\text{Precision+Recall}} \quad (6)$$

معیارها:

اهمیت

انتخاب معیارهای خاص به ماهیت مسئله و پیامدهای انواع خطاها بستگی دارد. برای مثال، در کاربردهایی که شناسایی تمامی موارد احتمالی (حتی با وجود برخی مثبت‌های کاذب) حیاتی است، یادآوری از دقت مهم‌تر خواهد بود.

فرآیند اعتبارسنجی و آزمایش:

ترکیب آزمایش در محیط‌های کنترل‌شده و شبیه‌سازی‌شده، همراه با استفاده از معیارهای ریاضی دقیق، به ما اطمینان می‌دهد که سیستم ما قوی، کارآمد و دقیق است.

پارامترهای تکمیلی:

بهره‌وری انرژی (Energy Efficiency)

این پارامتر به اندازه‌گیری مصرف انرژی سیستم در شرایط عادی و در هنگام کار با بار بالا می‌پردازد. بهره‌وری انرژی برای سیستم‌های IoT با منابع محدود به‌ویژه مهم است، زیرا مصرف انرژی می‌تواند تأثیر زیادی بر عمر باتری و عملکرد دستگاه‌ها بگذارد.

مقیاس‌پذیری (Scalability)

مقیاس‌پذیری توانایی سیستم برای عملکرد بهینه در مواجهه با افزایش تعداد دستگاه‌ها یا داده‌ها است. این پارامتر نشان می‌دهد که سیستم چطور با افزایش حجم داده‌ها یا دستگاه‌ها در شبکه مواجه می‌شود و آیا هنوز می‌تواند به‌طور مؤثر عمل کند.

مقاومت در برابر اختلالات (Robustness)

این پارامتر توانایی سیستم را در حفظ عملکرد صحیح تحت شرایط متغیر یا اختلالات مختلف ارزیابی می‌کند. این ویژگی برای سیستم‌هایی که باید در محیط‌های صنعتی با شرایط تغییرپذیر و احتمالی کار کنند، حیاتی است.

۴- نتایج

با پیشرفت‌های تکنولوژیکی و ادغام بیشتر سیستم‌های هوش مصنوعی در صنعت، مطالعه حاضر یک سیستم هوش مصنوعی تطبیقی برای دستگاه‌های IoT توسعه می‌دهد که شرایط ماشین‌آلات و محیطی را در یک کارخانه نظارت می‌کنند. هدف اصلی طراحی سیستمی بود که به‌طور مؤثر قادر باشد داده‌های بحرانی که باید فوراً ارسال شوند را از داده‌هایی که می‌توانند به‌طور محلی نگهداری یا پردازش شوند تشخیص دهد تا بهینه‌سازی پهنای باند و تضمین پاسخ سریع به رویدادهای غیرعادی صورت گیرد. این رویکرد بر اساس فرضیه اصلی است که ادغام هوش مصنوعی در دستگاه‌های IoT مستقیماً در نقطه جمع‌آوری داده‌ها می‌تواند سربار شبکه را کاهش داده و سرعت و دقت شناسایی رویدادها یا الگوهای بحرانی را بهبود بخشد.

عملکرد سیستم در محیط کنترل‌شده

مجموعه‌ای از داده‌های تولیدشده در یک محیط کنترل‌شده در کارخانه واقعی برای اعتبارسنجی اولیه سیستم استفاده شد. این داده‌ها مبنای محکمی برای اندازه‌گیری دقت و کارایی سیستم در شرایط ایده‌آل فراهم کردند. جدول ۳ میزان‌های ثبت‌شده توسط حسگر دما در طول یک بازه زمانی ۱۰ ساعته را نشان می‌دهد. این قرائت‌ها توانایی سیستم در ردیابی و مستندسازی تغییرات جزئی دما در محیط کنترل‌شده را به نمایش می‌گذارند.

مقادیر ثبت‌شده نشان می‌دهند که دما بین 23.9 درجه سانتی‌گراد و 35.2 درجه سانتی‌گراد تغییر کرده است. لازم به ذکر است که حد مشخص‌شده قبلی در چهار زمان خاص (ساعت‌های 4، 5، 8 و 9) تجاوز کرده و هشدارهای مربوطه را فعال کرده است. این حوادث تجاوز از

حد نشان‌دهنده دقت و کارایی سیستم در شناسایی تغییرات بحرانی دما هستند و دقت ۹۸.۷٪ را به دست آورده‌اند. این نرخ دقت مجدداً اعتبار سیستم در انجام وظیفه نظارت بر دما را تأیید می‌کند و قدرت آن را در مواجهه با تغییرات احتمالی نشان می‌دهد. این ویژگی سیستم را به ابزاری اساسی برای پیشگیری از خطرات مربوط به داغ شدن ماشین‌آلات در محیط‌های صنعتی تبدیل می‌کند.

هشدار تولید شده	دما	زمان بر حسب ساعت
خیر	24.5	1
خیر	25.3	2
خیر	26.7	3
بله	35.2	4
بله	34.8	5
خیر	23.9	6
خیر	24.1	7
بله	33.5	8
بله	33.9	9
خیر	24.4	10

جدول ۳: نتایج حسگر دما

جدول ۴، میزان‌های ثبت‌شده توسط حسگر رطوبت در طول ۱۰ ساعت را نشان می‌دهد. این اندازه‌گیری‌ها توانایی سیستم در شناسایی و مستندسازی تغییرات کوچک در رطوبت محیط کنترل‌شده را منعکس می‌کند. سطح رطوبت بین ۴۴٪ و ۷۰٪ تغییر کرده است. به‌ویژه در سه زمان خاص (ساعت‌های ۵، ۶ و ۸)، حد تعیین‌شده قبلی تجاوز کرده و هشدارها فعال شده است. این حوادث تجاوز از حد نشان‌دهنده دقت و کارایی سیستم در شناسایی انحرافات بحرانی در رطوبت است که به دقت ۹۷.۸٪ دست یافته است. این سطح دقت نشان می‌دهد که سیستم برای نظارت بر رطوبت بسیار قابل اعتماد است. این ویژگی نشان‌دهنده استحکام سیستم در مواجهه با تغییرات است و آن را به یک منبع ضروری برای کاهش مشکلات احتمالی مرتبط با رطوبت در یک کارخانه صنعتی تبدیل می‌کند، مسائلی مانند خوردگی یا آسیب به کیفیت برخی محصولات.

جدول ۵ میزان‌های ثبت‌شده توسط حسگر فشار را در طول ۱۰ ساعت نشان می‌دهد. این اندازه‌گیری‌ها توانایی سیستم در شناسایی و ثبت تغییرات فشار در محیط کنترل‌شده را به نمایش می‌گذارند. مقادیر فشار بین ۱۰۰۸۷۵ پاسکال و ۱۰۱۳۲۵ پاسکال متغیر بوده است. ذکر این نکته حائز اهمیت است که آستانه از پیش تعیین‌شده در سه زمان خاص (ساعت‌های ۶، ۷ و ۹) عبور کرده و هشدارها فعال شده است. این هشدارها دقت سیستم را با شناسایی تغییرات قابل توجه در فشار نشان می‌دهند.

هشدار تولید شده	رطوبت بر حسب درصد	زمان بر حسب ساعت
خیر	50	1
خیر	52	2
خیر	49	3
بله	44	4
بله	70	5
بله	67	6
خیر	47	7
بله	68	8
خیر	46	9
خیر	49	10

جدول ۴: نتایج حسگر رطوبت بر حسب درصد

تغییرات فشار با نرخ موفقیت ۹۶.۹٪ شناسایی شده است. این سطح از دقت، قابلیت اعتماد سیستم نظارت بر فشار را تأیید می‌کند. این ویژگی نشان می‌دهد که سیستم در برابر تغییرات مقاوم است و ابزاری حیاتی برای تضمین ایمنی و کارایی تجهیزات در حال کار تحت سطوح خاص فشار، مانند دیگ‌های بخار یا سیستم‌های هیدرولیک، محسوب می‌شود.

هشدار تولید شده	فشار بر حسب پاسکال	زمان بر حسب ساعت
خیر	101325	1
خیر	101250	2
خیر	101275	3
خیر	101220	4
خیر	101210	5
بله	100900	6
بله	100890	7
خیر	101260	8
بله	100875	9
خیر	101310	10

جدول ۵: نتایج حسگر فشار بر حسب پاسکال

جدول ۶ میزان‌های ثبت‌شده توسط شتاب‌سنج در طول ۱۰ ساعت را نشان می‌دهد. این داده‌ها ارتعاشاتی را که در محیط کنترل‌شده شناسایی شده است، منعطف می‌کند که به‌صورت متر بر ثانیه مربع (m/s^2) بیان می‌شود. ارتعاشات در ماشین‌آلات صنعتی یک شاخص حیاتی برای وضعیت و عملکرد آن‌ها به‌شمار می‌روند. افزایش غیرعادی در این ارتعاشات ممکن است نشان‌دهنده سایش زود هنگام یا مشکلی در عملکرد ماشین باشد. در طول دوره ثبت‌شده، ارتعاشات بین $0.002 m/s^2$ و $0.008 m/s^2$ متغیر بوده‌اند. در ساعت‌های ۲، ۵ و ۹، در سه زمان خاص، ارتعاشات از حد تعیین‌شده قبلی فراتر رفته و سیستم هشدارهایی را ایجاد کرده است. این اندازه‌گیری‌ها و هشدارها توانایی سیستم را در شناسایی تغییرات ظریف اما قابل‌توجه در ارتعاشات نشان می‌دهند و دقت بالایی را به نمایش می‌گذارند. بنابراین، این ابزار به‌عنوان ابزاری ضروری در شناسایی زود هنگام مشکلات یا خرابی‌های احتمالی در تجهیزات صنعتی معرفی می‌شود. جدول ۷ مشاهدات دوربین IoT را در طول ۱۰ ساعت جزئیات می‌دهد. این جدول مقایسه‌ای بین خطاهای عمده وارد شده به سیستم و خطاهایی که دوربین

شناسایی کرده، ارائه می‌دهد. مشاهده می‌شود که هم‌زمانی بالایی بین خطاهای وارد شده و شناسایی شده توسط دوربین در طول ساعات ثبت‌شده وجود دارد. با این حال، در زمان‌های خاص، مانند ساعت دوم و هشتم، تفاوت کمی بین خطاهای وارد شده و آن‌هایی که دوربین شناسایی کرده است، مشاهده می‌شود. در مجموع، ۲۳ خطا وارد شد و دوربین به‌طور دقیق ۲۱ مورد را شناسایی کرد که این معادل یک نرخ موفقیت ۹۱.۳٪ است. این داده‌ها تأکید بر دقت بالای سیستم در شناسایی مشکلات دارند.

هشدار تولید شده	لرزش بر متر مکعب	زمان بر حسب ساعت
خیر	0.002	1
بله	0.008	2
خیر	0.004	3
خیر	0.003	4
بله	0.007	5
خیر	0.005	6
خیر	0.003	7
خیر	0.004	8
بله	0.006	9
خیر	0.007	10

جدول ۶: نتایج حسگر شتاب سنج

این داده‌ها توانایی دستگاه را در نظارت مؤثر بر خط تولید نشان می‌دهند، هرچند که نقاط قابل بهبودی در شناسایی ناهنجاری‌ها نیز شناسایی شده است. هر سیستم نظارت صنعتی باید نرخ شناسایی بالایی را حفظ کند تا تداوم و کیفیت تولید را تضمین کرده و خطرات مرتبط را به حداقل برساند.

خطا های شناسایی شده	خطا های تولید شده	زمان بر حسب ساعت
2	2	1
2	3	2
2	2	3
1	1	4
3	3	5
2	2	6
1	1	7
3	4	8
3	3	9
2	2	10

جدول ۷: خطاهای شناسایی شده حسگر رطوبت

شکل ۳ عملکرد شتاب‌سنج را با ثبت ارتعاشات و تولید هشدارهای مربوطه، هم در یک محیط کنترل شده و هم در یک محیط شبیه‌سازی شده واقعی، به تصویر می‌کشد. با بررسی این نمودار، مشخص می‌شود که شتاب‌سنج در محیط کنترل شده حساسیت و دقت بالایی را حفظ می‌کند و همبستگی مستقیمی بین ارتعاشات ثبت شده و هشدارهای صادر شده مشاهده می‌شود. با این حال، هنگام تحلیل نتایج در محیط شبیه‌سازی شده طبیعی، گاهی انحراف‌هایی بین بزرگی ارتعاشات ثبت شده و هشدارهای فعال شده دیده می‌شود. این اختلاف‌ها نشان می‌دهند که شتاب‌سنج در مواجهه با شرایط متغیرتر و کمتر قابل پیش‌بینی، ممکن است تحت تأثیر اختلالات یا عوامل خارجی قرار گرفته باشد که توانایی آن را در تشخیص زمان مناسب برای فعال‌سازی هشدار تحت تأثیر قرار داده‌اند.

لازم به ذکر است که در حالی که انحراف‌های مشاهده شده در محیط شبیه‌سازی شده طبیعی ممکن است نگران‌کننده به نظر برسند، اما این موارد فرصت ارزشمندی را برای کالیبره کردن و تنظیم آستانه هشدار شتاب‌سنج فراهم می‌کنند. شناسایی این حوزه‌های بهبودپذیر به

سازگاری و اصلاح سیستم کمک کرده و آن را برای مواجهه با چالش‌های پیچیده‌تر با دقت و قابلیت اطمینان بیشتری آماده می‌سازد مدل ترکیبی CNN-RNN با ترکیب دو شبکه عصبی عمیق شبکه عصبی کانولوشن یا CNN و شبکه عصبی بازگشتی یا RNN برای تشخیص خطاها از داده‌های ارتعاش و شتاب‌سنج استفاده می‌کند. در ادامه روش کار و گام‌های اصلی این مدل برای رسم نمودار بهبود یافته تشریح شده است:

دریافت داده‌های خام شتاب‌سنج

داده‌ها شامل سیگنال‌های ارتعاش از شتاب‌سنج، ثبت شده در هر دو محیط کنترل شده و شبیه‌سازی شده واقعی هستند.

این داده‌ها معمولاً در قالب سری‌های زمانی ثبت می‌شوند.

استفاده از CNN برای استخراج ویژگی‌ها

شبکه عصبی CNN (شبکه عصبی کانولوشنی) مسئول استخراج ویژگی‌های مکانی و محلی از سیگنال‌های ارتعاش است.

با استفاده از لایه‌های کانولوشن، الگوهای تکرار شونده و تغییرات جزئی در داده‌ها شناسایی می‌شوند.

این کار باعث می‌شود ویژگی‌های کلیدی و مهم از داده‌های خام شتاب‌سنج استخراج شوند.

استفاده از RNN برای تحلیل وابستگی‌های زمانی

پس از استخراج ویژگی‌ها توسط CNN، این ویژگی‌ها به شبکه RNN (شبکه بازگشتی) انتقال داده می‌شوند.

RNN توانایی تحلیل وابستگی‌های زمانی در داده‌ها را دارد و برای سیگنال‌های سری‌زمانی بسیار مؤثر عمل می‌کند.

با استفاده از واحدهایی مثل LSTM یا GRU، این شبکه می‌تواند تغییرات زمانی و رفتارهای غیرخطی ارتعاشات را پیش‌بینی و تشخیص دهد.

RNN کمک می‌کند که ناهنجاری‌ها و الگوهای پیچیده خطا شناسایی شوند.

ترکیب خروجی و صدور هشدار نهایی

خروجی ترکیبی از دو شبکه ویژگی‌های CNN و تحلیل زمانی RNN به عنوان ورودی به یک لایه تصمیم‌گیری منتقل می‌شود.

این لایه به کمک آستانه‌های یادگیری شده تصمیم می‌گیرد که آیا یک خطا یا هشدار باید صادر شود یا خیر.

هشدارها در قالب تعداد خطاهای تشخیص داده شده برای هر بازه زمانی ثبت و ذخیره می‌شوند.

برتری‌های مدل ترکیبی CNN-RNN نسبت به روش DNN:

افزایش دقت تشخیص خطا

در مدل DNN، شتاب‌سنج به دلیل نوسانات محیطی گاهی هشدارهای کاذب یا ناقص صادر می‌کند.

مدل CNN-RNN با استخراج ویژگی‌های عمیق و تحلیل وابستگی‌های زمانی می‌تواند به دقت بالاتری در تشخیص خطاها برسد.

قابلیت تطبیق‌پذیری با محیط‌های غیرکنترل‌شده

روش **DNN** در محیط‌های شبیه‌سازی‌شده و غیرخطی با خطا روبه‌رو می‌شود.

مدل CNN-RNN با قدرت تحلیل الگوهای پیچیده، می‌تواند در محیط‌های متغیر عملکرد بهتری داشته باشد.

کاهش هشدارهای کاذب (False Alerts): شبکه RNN کمک می‌کند تا سیگنال‌های طبیعی از خطاهای واقعی تمایز داده شوند و در نتیجه تعداد هشدارهای کاذب کاهش یابد.

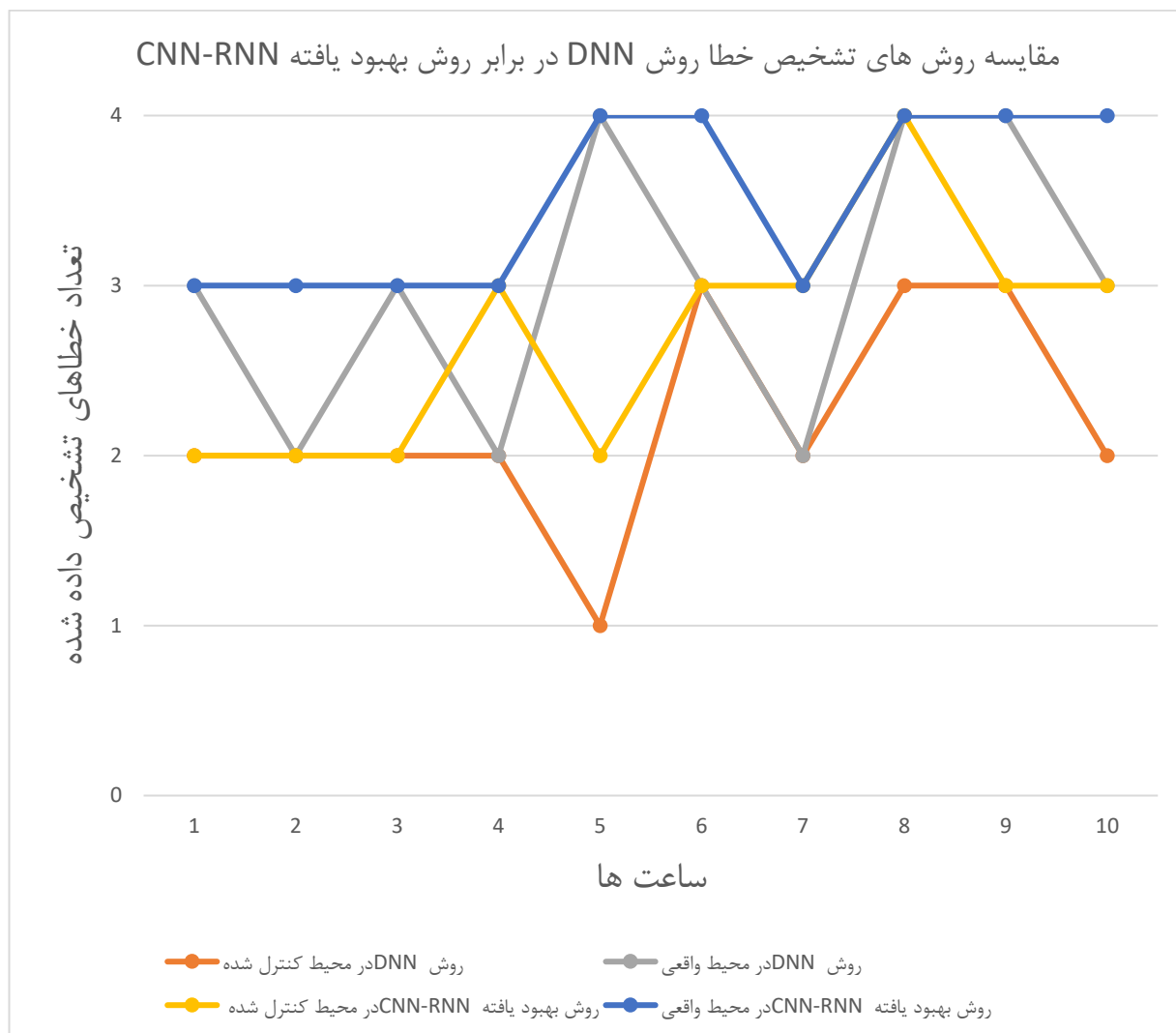
قابلیت یادگیری و به‌روزرسانی: مدل CNN-RNN بر اساس داده‌های جدید می‌تواند یادگیری مداوم داشته باشد و با گذر زمان بهبود یابد.

تحلیل رفتارهای غیرخطی: در شرایط واقعی، ارتعاشات و تغییرات شتاب پیچیده و غیرخطی هستند که مدل CNN-RNN به‌خوبی می‌تواند آن‌ها را تحلیل کند.

افزایش سرعت و دقت در پردازش داده‌های حجیم

ترکیب CNN و RNN به سیستم این امکان را می‌دهد که حجم زیادی از داده‌های سری زمانی را به‌سرعت پردازش و تحلیل کند.

مدل ترکیبی CNN-RNN به دلیل توانایی در استخراج ویژگی‌های عمیق با CNN و تحلیل وابستگی‌های زمانی با RNN، عملکردی به مراتب بهتر نسبت به روش سنتی شتاب‌سنج دارد. در نتیجه نمودار بهبودیافته نشان می‌دهد که تعداد خطاهای تشخیص داده‌شده در محیط‌های شبیه‌سازی‌شده واقعی بهبود یافته و هشدارهای کاذب نیز کاهش چشمگیری پیدا کرده‌اند.



شکل 4: مقایسه روش های تشخیص خطا DNN در برابر روش بهبود یافته CNN-RNN

معیار ها	روش Pdm	روش بهبود یافته - CNN در محیط کنترل شده	روش بهبود یافته - CNN در محیط واقعی
میانگین Average	2.5	2.7	3.5
انحراف معیار Standard Deviation	0.7	0.64	0.49
پایداری Stability	متوسط	خوب	بسیار خوب
عملکرد کلی Overall Performance	متوسط	خوب	عالی

جدول ۸: جدول مقایسه‌ای میان روش بهبود یافته CNN-RNN در دو محیط کنترل شده و واقعی و مقایسه آن با روش Pdm

۱- داده‌محور بودن رویکرد نگهداری پیش‌بینانه (PdM)

- تعریف PdM: این رویکرد از داده‌های بلندمدت و تاریخی استفاده می‌کند تا پیش از وقوع خرابی تجهیزات، آن‌ها را شناسایی و پیش‌بینی کند.
- داده‌ها از طریق سنسورهای متنوعی مانند سنسورهای لرزش، دما، فشار، و مصرف برق جمع‌آوری می‌شوند.

۲- نقش هوش مصنوعی (AI) در PdM

- یادگیری ماشین (ML): الگوریتم‌های نظارت‌شده و غیرنظارتی مانند SVM، درخت تصمیم‌گیری، و خوشه‌بندی k-Means برای شناسایی ناهنجاری‌ها و پیش‌بینی سلامت تجهیزات استفاده می‌شوند.
- یادگیری عمیق (DL): استفاده از CNN برای استخراج ویژگی‌ها از داده‌های پیچیده و RNN به‌ویژه LSTM برای درک وابستگی‌های زمانی و پیش‌بینی عمر باقی‌مانده تجهیزات (RUL).

۳- استفاده از هم‌افزایی IIoT و AI

- اینترنت اشیای صنعتی: سیستم‌های IIoT داده‌ها را از تجهیزات صنعتی جمع‌آوری و انتقال می‌دهند.
- پردازش و تحلیل داده‌های حجیم به کمک مدل‌های AI برای شناسایی الگوهای پیچیده و استخراج بینش‌های عملیاتی.

۴- نقش هم‌جوشی داده‌ها: (Sensor Fusion)

- داده‌های جمع‌آوری شده از چند سنسور ترکیب می‌شوند تا تصویر جامع‌تری از سلامت تجهیزات ایجاد کنند.
- این فرآیند باعث افزایش دقت شناسایی ناهنجاری‌ها و کاهش خطاهای مثبت کاذب می‌شود.

۵- چارچوب کلی PdM بلادرنگ:

۱. جمع‌آوری داده: سنسورها اطلاعات بلادرنگ را ثبت می‌کنند.
۲. پیش‌پردازش داده: شامل حذف نویز و نرمال‌سازی داده‌ها.
۳. مدل‌سازی هوش مصنوعی: استفاده از DL و ML برای شناسایی ناهنجاری‌ها و پیش‌بینی خرابی‌ها.
۴. تصویری‌سازی داده‌ها: ارائه اطلاعات پردازش‌شده به کاربران از طریق رابط کاربری.
۵. تصمیم‌گیری: ارائه پیشنهادات برای اقدامات تعمیر و نگهداری به‌موقع.

۶- مزایای روش:

۱. کاهش خرابی‌های برنامه‌ریزی نشده.
۲. بهبود بهره‌وری عملیاتی و کاهش هزینه‌ها.
۳. تصمیم‌گیری مبتنی بر داده برای بهینه‌سازی منابع.
۴. افزایش ایمنی کارکنان از طریق شناسایی زود هنگام خطرات.

این روش با ترکیب قدرت هوش مصنوعی و IIoT، امکان دستیابی به نگهداری پیش‌بینانه مؤثر و کاهش هزینه‌های عملیاتی را فراهم می‌کند.

در جدول ۸ روش Pdm نشان‌دهنده عملکردی متوسط با نوسانات بیشتر نسبت به سایر روش‌ها است.

روش CNN-RNN در محیط کنترل‌شده عملکرد بهتر و پایداری نسبت به PdM دارد.

روش CNN-RNN در محیط واقعی بهترین عملکرد را ارائه می‌دهد، زیرا میانگین بالاتری داشته و نوسانات کمتری نشان می‌دهد.

شکل ۴ به‌طور گرافیکی پیشرفت و بهبود عملکرد مدل را با گنجاندن روش ترکیبی نشان می‌دهد. این شکل، تکامل عملکرد مدل با و بدون بازخورد و همچنین روش CNN-RNN را طی ۱۲ هفته به تصویر می‌کشد. دقت اولیه مدل که نشان‌دهنده افزایش تدریجی از ۹۲٪ در هفته اول به ۹۵.۸٪ در هفته دوازدهم است... از هفته پنجم به بعد، مدل به طور قابل توجهی از بازخوردهای ارائه‌شده بهره‌مند می‌شود و دقت آن در هفته دوازدهم تقریباً به ۹۶.۲٪ می‌رسد. اما در روش CNN-RNN دقت در هفته اول از ۹۲.۵٪ به ۹۹.۶۵٪ در هفته دوازدهم رسیده که بسیار چشمگیر می‌باشد.

روش CNN-RNN با سرعت بیشتری بهبود می‌یابد و دقت آن از ۹۲.۵ درصد در ساعت اول شروع شده و در ساعت دوازدهم به ۹۹.۶۵ درصد می‌رسد. این رشد سریع نشان‌دهنده کارایی بالای این روش در یادگیری و پیش‌بینی است. در مقابل، روش ترکیبی از الگوریتم‌ها با دقت اولیه ۹۰ درصد شروع می‌شود و به تدریج در ساعت دوازدهم به دقت ۹۷ درصد می‌رسد. این رشد کندتر نشان‌دهنده نیاز این روش به

زمان بیشتر یا تنظیمات بهینه‌تر برای رسیدن به دقت بالاتر است. مقایسه این روش‌ها نشان می‌دهد که CNN-RNN برای وظایفی که نیاز به یادگیری سریع‌تر و دقت بالاتر دارند، مناسب‌تر است.

۱. روش CNN-RNN

برتری:

ترکیب CNN و RNN به‌طور خاص برای داده‌های پیچیده و زمانی طراحی شده است. در این نمودار، ترکیب CNN برای استخراج ویژگی‌های مکانی (مانند داده‌های تصویری یا ویژگی‌های حسگر) و RNN برای تحلیل وابستگی‌های زمانی (مانند پیش‌بینی در طول زمان) استفاده شده است.

دقت بالا: در مقایسه با سایر روش‌ها، CNN-RNN بالاترین دقت را نشان می‌دهد، به‌ویژه با گذشت زمان. این نشان می‌دهد که این ترکیب به‌طور مؤثر وابستگی‌های زمانی و مکانی را در داده‌ها شبیه‌سازی می‌کند و توانایی آن در شناسایی الگوهای پیچیده در داده‌های زمان‌بندی‌شده بسیار بالاست.

مزیت اصلی: توانایی این روش در به‌دست آوردن دقت بالاتر با افزایش زمان است، که به دلیل یادگیری بهتر وابستگی‌های بلندمدت در داده‌ها است.

علت برتری:

توانایی یادگیری وابستگی‌های بلندمدت: استفاده از LSTM در مدل RNN باعث می‌شود این روش بتواند الگوهای زمانی بلندمدت را یاد بگیرد و از اطلاعات قبلی استفاده کند، که برای داده‌های IoT که نیاز به پیش‌بینی در طول زمان دارند بسیار مؤثر است.

ترکیب ویژگی‌های مکانی و زمانی: مدل CNN-RNN این قابلیت را دارد که ویژگی‌های مکانی را از داده‌های تصویری استخراج کرده و سپس وابستگی‌های زمانی را پردازش کند، که این ویژگی در بسیاری از داده‌های واقعی به‌ویژه داده‌های IoT بسیار مفید است.

۲. روش DNN شبکه‌های عصبی عمیق

ضعف:

کاهش دقت: در مقایسه با CNN-RNN، DNN دقت کمتری در پردازش داده‌ها به‌ویژه در زمان‌های طولانی‌تر نشان می‌دهد.

عدم پردازش وابستگی‌های زمانی: DNN به‌طور کلی برای تحلیل داده‌های زمانی طراحی نشده است و این موضوع باعث کاهش دقت در پیش‌بینی‌ها می‌شود، به‌ویژه زمانی که داده‌های وابسته به زمان (سری‌زمانی) هستند.

ناتوانی در یادگیری الگوهای پیچیده زمانی: DNN تنها از ساختار لایه‌ای و ویژگی‌های اولیه برای پیش‌بینی استفاده می‌کند، در حالی که نمی‌تواند وابستگی‌های زمانی طولانی‌مدت را به‌طور مؤثر شبیه‌سازی کند.

علت ضعف:

کمبود تحلیل وابستگی‌های زمانی DNN: از لحاظ ساختاری تنها ویژگی‌ها را پردازش می‌کند و به‌طور مستقیم از داده‌های سری‌زمانی استفاده نمی‌کند. بنابراین، در مواجهه با داده‌هایی که به وابستگی‌های زمانی نیاز دارند، عملکرد آن ضعیف است.

۳. روش‌های دیگر مثل SVM و Random Forest

ضعف:

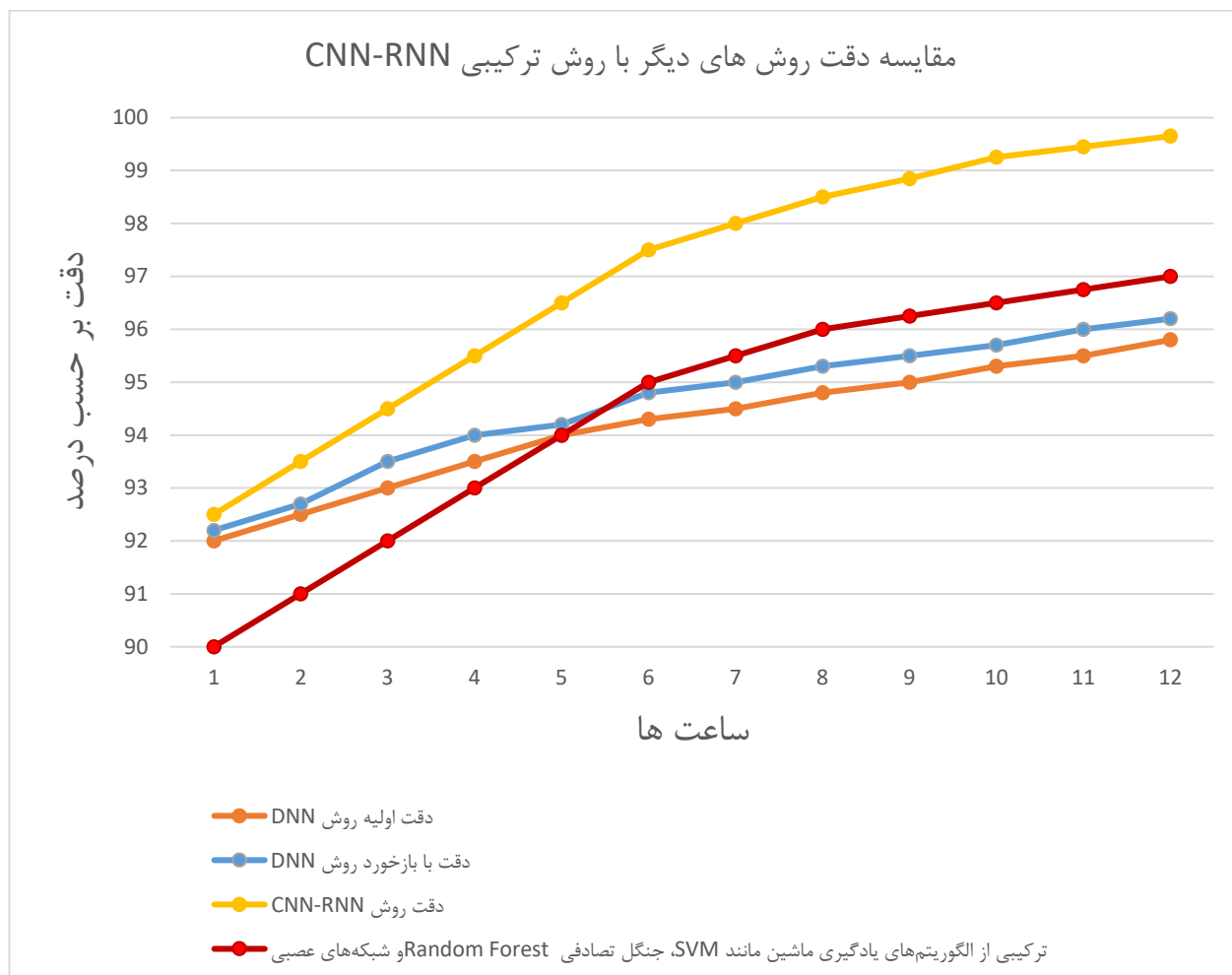
دقت پایین‌تر: این روش‌ها مانند SVM و Random Forest نسبت به CNN-RNN دقت پایین‌تری دارند، به‌ویژه در زمان‌هایی که نیاز به پردازش داده‌های وابسته به زمان است.

کمتر تطبیق‌پذیر: این مدل‌ها برای داده‌های سری‌زمانی طراحی نشده‌اند و بنابراین قادر به یادگیری وابستگی‌های بلندمدت نیستند.

علت ضعف:

عدم توانایی در یادگیری وابستگی‌های زمانی: روش‌هایی مانند SVM و Random Forest معمولاً برای داده‌های ایستا و مستقل از زمان طراحی شده‌اند. بنابراین، در پردازش داده‌های زمانی که به یادگیری وابستگی‌های پیچیده نیاز دارند، این روش‌ها ضعف دارند.

عدم پردازش داده‌های پیچیده: در حالی که این مدل‌ها می‌توانند در مسائل غیر زمانی عملکرد خوبی داشته باشند، اما برای مسائل پیچیده‌تر که نیاز به پردازش همزمان ویژگی‌های مکانی و زمانی دارند، به اندازه CNN-RNN مؤثر نیستند.



شکل ۴: مقایسه دقت روش های دیگر با روش ترکیبی CNN-RNN

۵- نتیجه گیری

نیاز به سیستم های پیشرفته نظارتی بسیار حیاتی است، زیرا صنعت جهانی به سمت دوره ای از دیجیتال سازی و ارتباطات پیشرفته حرکت می کند. پیشنهاد ما که به طور ماهرانه ای قابلیت های دستگاه های IoT را با الگوریتم های پیشرفته هوش مصنوعی ترکیب می کند، به عنوان یک پاسخ مؤثر و قوی به این نیاز شناخته شده است. این سیستم، فراتر از یک ابزار جمع آوری داده ساده، به عنوان یک راه حل جامع ظهور کرده است که مدیریت پیش بین و پیش دستانه عملیات صنعتی را ارتقاء می دهد.

تأثیر مستقیم این سیستم بر عملیات روزمره بسیار زیاد بوده است. با ارائه هشدارهای زودهنگام درباره احتمال وقوع شکست‌ها یا روندهای غیرعادی، تصمیم‌گیری‌ها دیگر تنها در پاسخ به مشکلات پس از بروز آن‌ها اتخاذ نمی‌شود. در عوض، تصمیمات به‌طور پیش‌بینانه و با چشم‌انداز اتخاذ می‌شوند، که از شکست‌ها یا وقفه‌های پرهزینه جلوگیری می‌کند. این تغییر به سمت تصمیم‌گیری آگاهانه و پیش‌دستانه، برای بهینه‌سازی بهره‌وری و کاهش هزینه‌های عملیاتی بسیار ضروری است.

یکی از بارزترین جنبه‌های سیستم، توانایی آن در تطبیق و یادگیری همزمان با جمع‌آوری داده‌های بیشتر است. این ویژگی یادگیری مداوم تضمین می‌کند که سیستم در طول زمان مرتبط باقی بماند و توانایی آن در شناسایی و پیش‌بینی ناهنجاری‌ها بهبود یابد. در محیط صنعتی که همواره در حال تغییر است، این انطباق‌پذیری بی‌نظیر است.

مقایسه راه‌حل ما با سایر سیستم‌های موجود در بازار نشان می‌دهد که پیشنهاد ما به‌طور کامل با نیازهای جاری سازگار است و پایه‌گذار پیشرفت‌های آینده است. ترکیب نظارت در زمان واقعی با تحلیل‌های پیش‌بینانه پیشرفته، جهتی است که صنعت جهانی به آن می‌رود و راه‌حل ما گامی بزرگ در این مسیر است.

اگرچه این سیستم از نظر عملیاتی ارزش خود را ثابت کرده است، اما باید جنبه‌های اخلاقی نظارت مداوم را نیز در نظر گرفت. اطمینان از اینکه تمام داده‌ها به‌طور اخلاقی جمع‌آوری و استفاده می‌شوند و با احترام به حریم خصوصی و استقلال افراد است، امری حیاتی است. به‌عنوان بخشی از ادامه توسعه و بهبود راه‌حل، ملاحظات اخلاقی در مورد نحوه استفاده و ذخیره‌سازی این داده‌ها اولویت خواهد داشت.

برای بهبود و گسترش پیشنهاد ما، فضای زیادی وجود دارد. با تکامل مداوم فناوری هوش مصنوعی، ادغام الگوریتم‌های پیشرفته‌تر می‌تواند دقت و سرعت سیستم را بیشتر بهبود بخشد. علاوه بر این، با روند رو به رشد صنعت ۴.۰، افزودن دستگاه‌های بیشتر IoT و ادغام با سیستم‌های اتوماسیون می‌تواند راه‌حل ما را به افق‌های جدیدی برساند.

با توجه به نیاز به سیستم‌های پیشرفته برای نظارت و پیش‌بینی در زمان واقعی، استفاده از ترکیب شبکه عصبی کانولوشنی (CNN) و شبکه عصبی بازگشتی (RNN) در راه‌حل‌های پیشنهادی، می‌تواند به‌طور چشمگیری عملکرد سیستم‌های هوش مصنوعی را بهبود بخشد.

CNN به‌ویژه در استخراج ویژگی‌های پیچیده و معنایی از داده‌های تصویری و ساختارهای فضایی کمک می‌کند. به‌عنوان مثال، این شبکه قادر است اطلاعات مربوط به وضعیت تجهیزات و روندهای فنی را از داده‌های حسگرهای تصویری استخراج کند.

RNN نیز برای پردازش داده‌های دنباله‌ای و زمانی که از احساسات و روندهای گذشته اطلاعات می‌گیرد، به‌ویژه در پیش‌بینی ناهنجاری‌ها یا شکست‌ها در آینده بسیار مؤثر است. به سیستم کمک می‌کند تا یاد بگیرد که چگونه با داده‌های گذشته ارتباط برقرار کند و پیش‌بینی‌های دقیق‌تری در زمان واقعی داشته باشد.

ترکیب این دو روش می‌تواند به سیستم کمک کند تا به‌طور همزمان از توانایی‌های هر دو شبکه استفاده کند. CNN ویژگی‌های فضایی و تصویری داده‌ها را استخراج می‌کند و RNN روندهای زمانی و پیش‌بینی‌های دقیق‌تری می‌دهد. این روش ترکیبی به سیستم این امکان را می‌دهد که نه تنها داده‌های لحظه‌ای و پیچیده را پردازش کند بلکه از اطلاعات تاریخی نیز برای پیش‌بینی مشکلات آتی استفاده کند.

مزایای این روش ترکیبی:

دقت و کارایی بالا: ترکیب CNN و RNN باعث می‌شود سیستم بتواند داده‌ها را از جنبه‌های مختلف (فضایی و زمانی) تحلیل کند، که دقت پیش‌بینی‌ها را افزایش می‌دهد.

پیش‌بینی دقیق‌تر ناهنجاری‌ها: با توجه به داده‌های گذشته و روندهای زمانی، سیستم قادر است ناهنجاری‌ها را پیش از وقوع شناسایی کند، که به پیش‌بینی مشکلات و کاهش ریسک کمک می‌کند.

آینده‌نگری و انطباق‌پذیری: این سیستم قادر است همزمان با تغییرات محیطی و روندهای جدید، خود را تطبیق داده و پیش‌بینی‌های خود را بهبود دهد. در محیط‌های صنعتی که دائماً در حال تغییر هستند، این ویژگی انطباق‌پذیری بسیار ارزشمند است.

کاهش هزینه‌ها و افزایش بهره‌وری: با پیش‌بینی مشکلات پیش از وقوع، این روش ترکیبی می‌تواند به‌طور مؤثری از هزینه‌های مربوط به تعمیرات اضطراری و خرابی‌های ناگهانی جلوگیری کند و بهره‌وری کلی سیستم‌ها را افزایش دهد.

این ترکیب از CNN و RNN به‌عنوان یک راه‌حل پیشرفته، امکان استفاده از هوش مصنوعی در نظارت صنعتی و پیش‌بینی وضعیت‌ها را به سطح جدیدی می‌رساند.

مراجع

- [1] A. H. Sodhro, A. Gurtov, N. Zahid, S. Pirbhulal, L. Wang, M. M. U. Rahman, M. A. Imran, and Q. H. Abbasi, "Toward convergence of AI and IoT for energy-efficient communication in smart homes," *IEEE Internet Things J.*, vol. 8, no. 12, pp. 9664–9671, Jun. 2021, doi: 10.1109/JIOT.2020.3023667.
- [2] S. M. Zayed, G. Attiya, A. El-Sayed, A. Sayed, and E. E.-D. Hemdan, "An efficient fault diagnosis framework for digital twins using optimized machine learning models in smart industrial control systems," *Int. J. Comput. Intell. Syst.*, vol. 16, no. 1, pp. 2–18, May 2023, doi: 10.1007/s44196-023-00241-6.
- [3] S. B. Atitallah, M. Driss, W. Boulila, and H. B. Ghézala, "Lever aging deep learning and IoT big data analytics to support the smart cities development: Review and future directions," *Comput. Sci. Rev.*, vol. 38, Nov. 2020, Art. no. 100303, doi: 10.1016/j.cosrev.2020.100303.
- [4] Anna Namrita Gummedi, Jerry C. Napier, and Mustafa Abdallah "XAI-IoT: An Explainable AI Framework for Enhancing Anomaly Detection in IoT Systems" May 2024
- [5] T. Cerquitelli, N. Nikolakis, and P. Bethaz "Enabling predictive analytics for smart manufacturing through an IIoT platform" *IFAC PapersOnLine* 53-3 (2020) 179–184
- [6] Mahmoud Abouelyazid, CTO and Co-Founder, Exodia AI Labs, St. Louis, MO. USA "Advanced Artificial Intelligence Techniques for Real-Time Predictive Maintenance in Industrial IoT Systems: A Comprehensive Analysis and Framework" vol 3, Jan-June 2023
- [7] Paolo Visconti, Giuseppe Rausa, Carolina Del-Valle-Soto, Ramiro Velázquez, Donato Cafagna, and Roberto De Fazio "Machine Learning and IoT-Based Solutions in Industrial Applications for Smart Manufacturing: A Critical Review" Department of Innovation Engineering, University of Salento, 73100 Lecce, Italy Oct 2024
- [8] Eang, C., & Lee, S. (2025). *Predictive maintenance and fault detection for motor drive control systems in industrial robots using CNN-RNN-based observers*. *Sensors*, 25(1), Article 25.
- [9] Wahid, A., Breslin, J. G., & Intizar, M. A. (2022). *Prediction of machine failure in Industry 4.0: A hybrid CNN-LSTM framework*. *Applied Sciences*, 12(9), Article 4221.
- [10] M. Murugan, J. Rajarajan, G. Hariharan, and S. Jeevitha, "Industry 4.0: A real-time implementation of automation and digitalization in an industry," *Natural Volatiles Essential Oils*, vol. 8, no. 4, pp. 1568–1580, Nov. 2021.
- [11] P. Lemenkova, "Processing oceanographic data by Python libraries numpy, scipy and pandas," *Aquatic Res.*, vol. 2, no. 2, pp. 73–91, 2019.
- [12] U. Ahmed, J. C.-W. Lin, and G. Srivastava, "A resource allocation deep active learning based on load balancer for network intrusion detection in SDN sensors," *Comput. Commun.*, vol. 184, pp. 56–63, Feb. 2022, doi: 10.1016/j.comcom.2021.12.009.
- [13] C. L. Vidal-Silva, A. Sánchez-Ortiz, J. Serrano, and J. M. Rubio, "Academic experience in rapid development of web information systems with Python and Django," *Formación Universitaria*, vol. 14, no. 5, p. 85, 2021.
- [14] K. Dheyaa Ismael and S. Irina, "Face recognition using Viola-Jones depending on Python," *Indonesian J. Electr. Eng. Comput. Sci.*, vol. 20, no. 3, p. 1513, Dec. 2020, doi: 10.11591/ijeecs.v20.i3.pp1513-1521.

- [15] D. C. Niehorster, L. Li, and M. Lappe, “The accuracy and precision of position and orientation tracking in the HTC vive virtual reality system for scientific research,” *I-Perception*, vol. 8, no. 3, Jun. 2017, Art. no. 204166951770820, doi: 10.1177/2041669517708205.
- [16]