

نظارت هوشمند در اینترنت اشیا با استفاده از یادگیری عمیق برای تحلیل و پیش بینی بلادرنگ

سینا نصراللهی^۱، سید ابراهیم دشتی^۲^۱ دانشکده فنی، واحد شیراز، دانشگاه آزاد اسلامی، شیراز، ایران^۲ گروه برق و کامپیوتر، واحد جهرم، دانشگاه آزاد اسلامی، جهرم، ایران

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 مدل های ترکیبی CNN و RNN، در نظارت بلادرنگ و تحلیل پیش بینانه پیشنهاد شده است. این روش سیستمی نوین ارائه می دهد که داده های حسگرهای IoT را با استفاده از CNN برای تحلیل ویژگی های مکانی و RNN برای پردازش توالی های زمانی ترکیب می کند. این سیستم ناهنجاری ها را شناسایی کرده و خرابی های احتمالی را پیش بینی می کند، و امکان مداخلات پیشگیرانه را برای کاهش هزینه های عملیاتی و افزایش کارایی فراهم می آورد. ارزیابی سیستم با مجموعه داده هایی 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 نشان دهنده توانایی سیستم در پیش بینی دقیق خرابی های بحرانی است.

دریافت مقاله: ۱۸/۱۰/۱۴۰۳

پذیرش مقاله: ۱۰/۰۶/۱۴۰۴

کلمات کلیدی: اینترنت اشیا، یادگیری عمیق، شبکه عصبی پیچشی،
شبکه عصبی بازگشتی، شناسایی بلادرنگ ناهنجاری

۱- مقدمه

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 [۱][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، نه‌تنها 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 نه‌تنها 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 اشیاء ارائه می‌دهد پیشنهاد ما در زمینه

سیستم تطبیقی مبتنی بر هوش مصنوعی، یک پیشرفت قابل توجه است و راه حلی پویا و تکاملی برای چالش مدیریت داده در دستگاه های اینترنت اشیا ارائه می دهد در ادامه چند روش مشابه دیگر را ذکر شده است.

روش اول هدف ارائه یک چارچوب هوش مصنوعی قابل تفسیر (XAI) برای تشخیص ناهنجاری ها در سیستم های اینترنت اشیا (IoT) طراحی شده است. چارچوب پیشنهادی از روش های پیشرفته XAI استفاده می کند تا علاوه بر افزایش دقت در شناسایی ناهنجاری ها، شفافیت و اعتمادپذیری مدل های هوش مصنوعی را نیز ارتقا دهد. در این راستا، ابزارهایی مانند SHAP (SHapley Additive) (exPlanations) برای تحلیل اهمیت ویژگی ها و تأثیر آن ها بر پیش بینی مدل و LIME (Local Interpretable Model-Agnostic Explanations) به منظور تفسیر محلی پیش بینی های خاص مورد استفاده قرار گرفته اند. علاوه بر این، کتابخانه های پایتون از جمله Keras برای ساخت و آموزش مدل ها، Pandas برای پردازش داده ها و Scikit-learn برای محاسبات و ارزیابی های آماری استفاده شده اند.

مجموعه داده های به کاررفته در این مطالعه شامل داده های MEMS از حسگرهای مبتنی بر تولید هوشمند و داده های N-BaIoT مرتبط با حملات باتنت در سیستم های IoT است. اگرچه چارچوب پیشنهادی مزایای بسیاری دارد، اما با چالش هایی مانند پیچیدگی و هزینه های محاسباتی بالا، وابستگی به داده های آموزشی باکیفیت، محدودیت در مقیاس پذیری، و دشواری در استفاده به صورت زمان واقعی مواجه است. همچنین، تفسیر برخی نتایج این چارچوب در شرایط خاص می تواند پیچیده باشد.

از جمله مزایای این چارچوب می توان به افزایش دقت در تشخیص ناهنجاری ها، تطبیق پذیری با داده های واقعی، تحلیل اهمیت ویژگی ها و افزایش شفافیت و اعتمادپذیری مدل ها اشاره کرد. این سیستم با استفاده از پارامترهای عملکردی نظیر Precision، Recall، و F1-score ارزیابی شده و توانایی بهبود عملکرد مدل های هوش مصنوعی در شناسایی ناهنجاری ها و تفسیر قابل اعتماد آن ها را نشان داده است.

نوآوری اصلی این روش، ترکیب روش های XAI برای افزایش شفافیت و ارائه چارچوبی است که علاوه بر تشخیص ناهنجاری ها، تحلیل اهمیت ویژگی ها را به شکلی دقیق و کاربردی امکان پذیر می سازد. تمرکز این مطالعه بر بهبود اعتمادپذیری و کارایی سیستم های IoT از طریق استفاده از هوش مصنوعی قابل تفسیر است [4].

روش دوم بررسی و پیاده سازی روش های تحلیل پیش بینانه برای سیستم های 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 شبیه سازی در محیط های رباتیک استفاده می کند. داده های مورد استفاده در این تحقیق عمدتاً شامل اطلاعات جمع آوری شده از حسگرهای صنعتی در کارخانه های تولیدی و سیستم های مبتنی بر IoT صنعتی هستند.

با وجود مزایای متعدد، این سیستم با چالش هایی همچون پیچیدگی پیاده سازی، نیاز به منابع محاسباتی بالا، وابستگی به داده های کیفی و متنوع، مدیریت پیچیده داده ها و تنظیمات اولیه، محدودیت در پیش بینی کامل خرابی ها، و چالش های مرتبط با امنیت و حریم خصوصی داده ها مواجه است. با این حال، مزایای استفاده از این پلتفرم شامل افزایش قابلیت پیش بینی و کاهش خرابی های غیرمنتظره، انعطاف و تطبیق پذیری با سیستم های مختلف، امکان تجزیه و تحلیل داده های حجیم و لحظه ای، قابلیت تجمیع داده های چنددوره ای، و پشتیبانی از تجسم داده و مدیریت ساده تر تعمیر و نگهداری است.

پلتفرم SERENA با بهره گیری از تحلیل دوره های 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 داده ها استفاده می کند. همچنین، پارامترهایی مانند Aggregation، Horizon، و Split size در فرآیند تحلیل داده های حسگرها به کار گرفته شده اند.

نوآوری این تحقیق در استفاده از پلتفرم SERENA و تحلیل پیش بینانه دوره های زمانی نهفته است که امکان شناسایی تغییرات تدریجی در سیستم های صنعتی و پیشگیری از خرابی های غیرمنتظره را فراهم می کند [5]. در این روش تأکید بر توانمندسازی تولید هوشمند از طریق تحلیل پیش بینانه و استفاده از پلتفرم های 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 می‌کند. 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 قابل توجهی 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 (IoT) 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 (IoT) در کاربردهای صنعتی است که می‌تواند نقش قابل توجهی 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 LSTM و CNN تکی - معماری سبک‌تر از CNN- LSTM-مناسب برای اجرا در زمان واقعی	Predictive Maintenance Dataset ≈ ۱۰,۰۰۰ رکورد، داده حسگر موتور صنعتی	CNN-RNN برای تشخیص خطا و نگهداشت پیشگويانه	Sensors, 2024 (Predictive Maintenance of Robot Motors)	1

2	Applied Sciences, 2022 (Industry 4.0 Case Study)	CNN-LSTM با <i>temporal skip connection</i> پیش بینی خرابی ماشین	دیتاست موردی Microsoft سنسورهای صنعتی چندمتغیره	-مدل سازی بهتر الگوهای فصلی و بلندمدت- بهبود معیارهای MAE، RMSE و R^2 نسبت به مدل های ساده	-هزینه محاسباتی بالا- دشواری استفاده در دستگاه های IoT با منابع محدود
---	--	--	--	--	---

جدول ۱: دو تحقیق مشابه روش ترکیبی CNN/RNN

جدول ۱ دو پژوهش مشابه در زمینه ترکیب شبکه های CNN و RNN/LSTM در کاربردهای IoT و صنعت ۴,۰ ارائه می دهد. مطالعه نخست [8] از ترکیب CNN و RNN برای تشخیص خطا در ربات های صنعتی استفاده کرده است. مزیت اصلی آن سادگی معماری و سرعت بالای پردازش بلادرنگ است، اما محدودیت آن در شناسایی وابستگی های زمانی بلندمدت باقی مانده است. مطالعه [9] با بهره گیری 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 پیاده سازی روی دستگاه های IoT دارد.

پژوهش دوم، بر اجرای بهینه روی دستگاه های IoT با منابع محدود تمرکز شده است؛ بدین منظور تکنیک هایی نظیر Quantization و Pruning به کار گرفته شده تا مدل سبک تر و مناسب برای پردازش بلادرنگ گردد. بنابراین، کار حاضر تمایز خود را در حوزه ی کاربردی (اجرا روی دستگاه های IoT با محدودیت منابع) آشکار می سازد.

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 $[0, 1]$ 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.

برچسب گذاری داده ها: داده های ناهنجار با استفاده از قوانین مشخص یا مقادیر آستانه ای (Threshold) تعریف شده به عنوان "رخداد ناهنجاری" برچسب گذاری شدند.

پیش پردازش داده های تصویری

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

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

افزایش داده (Data Augmentation):

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

برش تصادفی (Random Cropping): بخش هایی از تصاویر به طور تصادفی انتخاب و بریده شدند.

تغییر شدت روشنایی: شدت نور تصاویر برای شبیه سازی شرایط واقعی تغییر یافت.

معکوس کردن افقی و عمودی: تصاویر به طور تصادفی برعکس شدند. این تکنیک ها باعث افزایش تنوع داده های آموزشی شده و از بیش برآزش (Overfitting) جلوگیری می کند.

تشخیص و حذف مقادیر گمشده (Missing Values)

مقادیر گمشده در داده های سری زمانی با استفاده از روش های میانگین گیری یا برون یابی خطی (Linear Interpolation) جایگزین شدند. در مواردی که مقادیر زیادی از داده ها گمشده بودند، نمونه ها 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 (کرنل) است.

b : بایاس است.

فرمول RNN (۲)

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

h_t خروجی در زمان t است.

x_t ورودی در زمان t است.

W_h وزن های لایه ها هستند.

$W_h h$ وزن های 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 محیط های صنعتی طراحی شده است. با رشد صنعت ۴,۰ [11] کارایی، ایمنی، و بهینه سازی فرآیندهای تولید به عوامل حیاتی تبدیل شده اند. بنابراین، وجود یک سیستم قوی و قابل اعتماد که بتواند اطلاعات دقیق و به موقعی از وضعیت ماشین آلات و شرایط محیطی ارائه دهد، بسیار ضروری است.

برای این منظور، چندین حسگر استفاده شده اند:

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

حسگر رطوبت: برای نظارت بر سطح رطوبت محیط های صنعتی استفاده می شود، که از خوردگی ماشین آلات و آسیب به محصولات حساس به رطوبت جلوگیری می کند.

حسگر فشار: برای نظارت بر ماشین آلاتی که در شرایط فشار خاصی کار می کنند (مانند دیگ های بخار یا سیستم های هیدرولیک).

شتاب سنج سه محوره: ارتعاشات غیرعادی ماشین آلات را تشخیص می دهد که ممکن است نشانه ای از سایش، نقص یا نیاز به تعمیرات پیشگیرانه باشد.

دوربین اینترنت اشیا: نظارت تصویری از خط تولید ارائه می دهد تا موانع، خرابی ها، یا مشکلات دیگر به صورت بلادرنگ شناسایی شوند.

این حسگرها به طور استراتژیک در ماشین آلات حیاتی و نقاط خاص محیط تولید قرار داده شده اند. داده های جمع آوری شده توسط این حسگرها به سرور مرکزی منتقل می شود، جایی که مدل هوش مصنوعی آن ها را در بلادرنگ پردازش و تحلیل می کند. در صورت شناسایی هرگونه ناهنجاری، هشدارهایی برای کارکنان کارخانه ارسال می شود تا اقدامات اصلاحی انجام دهند.

۲-۲- مجموعه داده ها

برای توسعه، پیاده سازی، و اعتبارسنجی سیستم پیشنهادی، دو منبع اصلی برای جمع آوری داده ها استفاده شده است محیط های کنترل شده برای شبیه سازی شرایط خاص و محیط های تولید صنعتی برای جمع آوری داده های واقعی. این مجموعه داده ها به طور خاص طراحی شده اند 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.

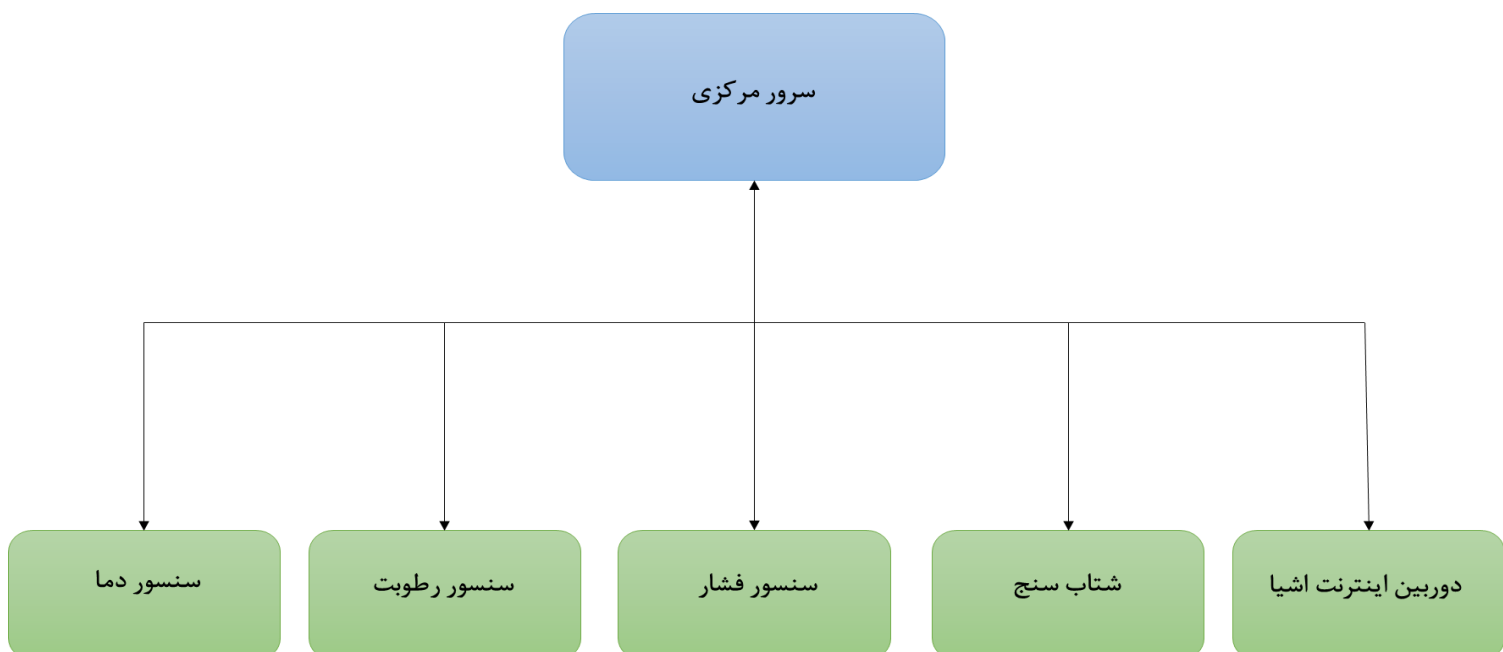
تصاویر با مشکلات مشخص (مانند انسداد دید) به دسته ناهنجاری اضافه شدند.

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

تشخیص و حذف داده های پرت

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

مدیریت داده های گمشده

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

ابزارهای مورد استفاده

Pandas برای مدیریت داده ها، پاک سازی و انجام تحلیل های آماری ابتدایی [10][11].

Scikit-learn برای نرمال سازی داده ها و انجام تکنیک های پیش پردازش پیشرفته مانند استاندارد سازی و رمز گذاری One-Hot.

تبدیلات یا نرمال سازی های اعمال شده

استاندارد سازی: برای داده های عددی، از استاندارد سازی استفاده شد تا ویژگی ها میانگین صفر و انحراف معیار یک داشته باشند. این کار برای اطمینان از همگرایی بهتر الگوریتم های یادگیری عمیق [12] و مقایسه مؤثر ویژگی ها در مدل های ترکیبی CNN-RNN ضروری است.

نرمال سازی Min-Max: برای ویژگی های عددی که دامنه های بسیار متفاوتی دارند، از نرمال سازی Min-Max برای تبدیل مقادیر به بازه [۰, ۱] استفاده شد. این تکنیک کمک می کند تا رابطه بین مقادیر حفظ شود و داده ها به طور یکنواخت مقیاس بندی شوند.

رمز گذاری One-Hot

ویژگی های دسته ای، مانند نوع دستگاه یا شرایط محیطی، با استفاده از روش One-Hot Encoding به بردارهای باینری تبدیل شدند. این تکنیک برای پردازش داده های غیر عددی که توسط مدل های یادگیری عمیق قابل استفاده نیستند، ضروری است.

آماده سازی داده ها برای مدل های ترکیبی CNN-RNN

داده های تصویری: تصاویر جمع آوری شده از دوربین های IoT ابتدا با استفاده از نرمال سازی رنگ و تغییر اندازه به ابعاد ثابت (۲۲۴×۲۲۴ 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.

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

جدول ۲: خلاصه روش های پیش پردازش استفاده شده

۱-۳- روش شناسی 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 تصویربرداری 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 به‌روزرسانی 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 اشیاء 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):

ویژگی های استخراج شده توسط CNN و RNN در این مرحله با هم ترکیب می شوند. این ترکیب به مدل اجازه می دهد از اطلاعات مکانی و زمانی به طور هم زمان استفاده کند و یک نمای کلی تر و جامع تر از داده ها به دست آورد.

لایه تصمیم گیری (Decision Layer):

در این بخش، داده های ترکیبی پردازش شده وارد یک لایه کاملاً متصل (Fully Connected Layer) می شوند که وظیفه نهایی سازی پیش بینی یا تصمیم گیری را دارد [14]. نتایج این لایه می تواند شامل دسته بندی، پیش بینی سری زمانی یا اجرای یک عملکرد خاص در دستگاه IoT باشد.

خروجی نهایی (Output):

این بخش نتیجه نهایی را در قالبی مانند پیش بینی دسته بندی، پیش بینی مقدار، یا فعال سازی عملکرد دستگاه ارائه می دهد.

مزایای روش ترکیبی CNN و RNN نسبت به DNN:

کاهش پیچیدگی محاسباتی:

برخلاف DNN، ترکیب CNN و RNN از ساختارهای تخصصی برای پردازش داده ها استفاده می کند که محاسبات کمتری را به همراه دارد و برای دستگاه های IoT با منابع محدود مناسب تر است.

بهره برداری از ویژگی های مکانی و زمانی:

CNN برای داده های مکانی (تصویری) و RNN برای داده های زمانی (سری زمانی یا وابسته به توالی) طراحی شده اند. ترکیب این دو شبکه امکان استفاده هم زمان از اطلاعات زمانی و مکانی را فراهم می کند، در حالی که DNN در شناسایی این ویژگی ها محدودتر عمل می کند.

کاهش اندازه مدل:

DNN به دلیل تعداد زیاد لایه ها و وزن ها معمولاً حجیم تر است. حذف DNN و استفاده از CNN و RNN باعث می شود که مدل نهایی سبک تر باشد و با حافظه کمتر (مانند SRAM) سازگار باشد.

عملکرد بهتر در زمان واقعی (Real-Time):

روش ترکیبی CNN و RNN با بهینه سازی داده ها و پردازش آن ها در دسته های مرتبط، عملکرد بهتری در پردازش داده ها به صورت بلادرنگ دارد. این موضوع برای دستگاه های IoT که نیازمند پاسخ سریع هستند، حیاتی است.

انعطاف پذیری بیشتر در طراحی مدل:

در این روش، می توان از CNN و RNN به صورت موازی یا تریبی استفاده کرد و آن ها را با معماری های مختلف ترکیب کرد تا دقت و کارایی مدل افزایش یابد. این انعطاف در DNN کمتر است.

صرفه جویی در مصرف انرژی:

دستگاه های IoT به دلیل محدودیت انرژی، نیازمند مدل های سبک تر و کم مصرف تر هستند. حذف DNN و استفاده از CNN و RNN که بهینه تر عمل می کنند، مصرف انرژی را کاهش می دهد.

بهینه سازی ها برای عملکرد بهتر:

با توجه به محدودیت های سخت افزاری دستگاه های IoT، چندین تکنیک به منظور تضمین عملکرد بلا درنگ به کار گرفته شد:

هرس کردن (Pruning): حذف وزن ها یا نورون هایی که تأثیر کمی بر دقت دارند. این تکنیک باعث کاهش اندازه مدل و افزایش سرعت استنتاج می شود.

کمیت سازی (Quantization): تبدیل وزن ها و بایاس های مدل از فرمت ۳۲ بیت به فرمت های کوچکتر مانند ۸ بیت. این تکنیک باعث کاهش حجم مدل و افزایش سرعت استنتاج به دلیل کاهش بار محاسباتی می شود.

کامپایل مدل در زمان اجرا (On-the-Fly Compilation): استفاده از کامپایلر های تخصصی برای ترجمه مدل به دستورالعمل های مخصوص دستگاه و بهینه سازی عملکرد آن.

بهینه سازی حافظه: بارگذاری مدل در سریع ترین حافظه موجود روی دستگاه مانند SRAM به جای حافظه فلش یا خارجی برای تسریع در فرآیند پردازش.

پردازش دسته ای (Batch Processing): تا حد امکان داده ها به صورت دسته ای پردازش شدند تا از قابلیت های موازی سازی دستگاه بهره برداری شود و کارایی سیستم بهبود یابد.

این بهینه سازی ها اطمینان می دهند که مدل به طور مؤثر و در زمان واقعی روی دستگاه IoT اجرا شده و استنتاج ها به سرعت و دقت انجام می شوند.

۳-۳- اعتبار سنجی و آزمایش

برای اطمینان از صحت عملکرد سیستم، به ویژه در محیط های صنعتی با شرایط متغیر، آزمایش های مختلفی طراحی و اجرا شد:

آزمایش در محیط کنترل شده

در این سناریو، دستگاه IoT با مدل یکپارچه در محیطی آزمایشگاهی یا مشابه آن آزمایش می شود. در این شرایط، تمام متغیرها و شرایط به طور دقیق کنترل و نظارت می شوند. این نوع آزمایش امکان ارزیابی عملکرد سیستم در شرایط ایده آل را فراهم می کند. به عنوان مثال، می توان مقادیر خاص دما و رطوبت را شبیه سازی کرده و بررسی کرد که آیا دستگاه پاسخ های صحیح و مورد انتظار را ارائه می دهد یا خیر. این آزمایش برای شناسایی مشکلات بنیادی یا خرابی های آشکار کاربرد دارد.

آزمایش در محیط شبیه سازی شده واقعی

در این نوع آزمایش، دستگاه در محیطی شبیه سازی می شود که شرایط واقعی را برای سیستم فراهم می آورد، به طوری که دستگاه در شرایط نزدیک به واقعیت کار کند. این آزمایش برای ارزیابی عملکرد سیستم در مواجهه با شرایط پیچیده تر و پویاتر طراحی شده است. در این محیط، نظارت و کنترل همچنان وجود دارد، اما دستگاه باید توانایی تعامل با شرایط متغیر و غیر قابل پیش بینی را داشته باشد.

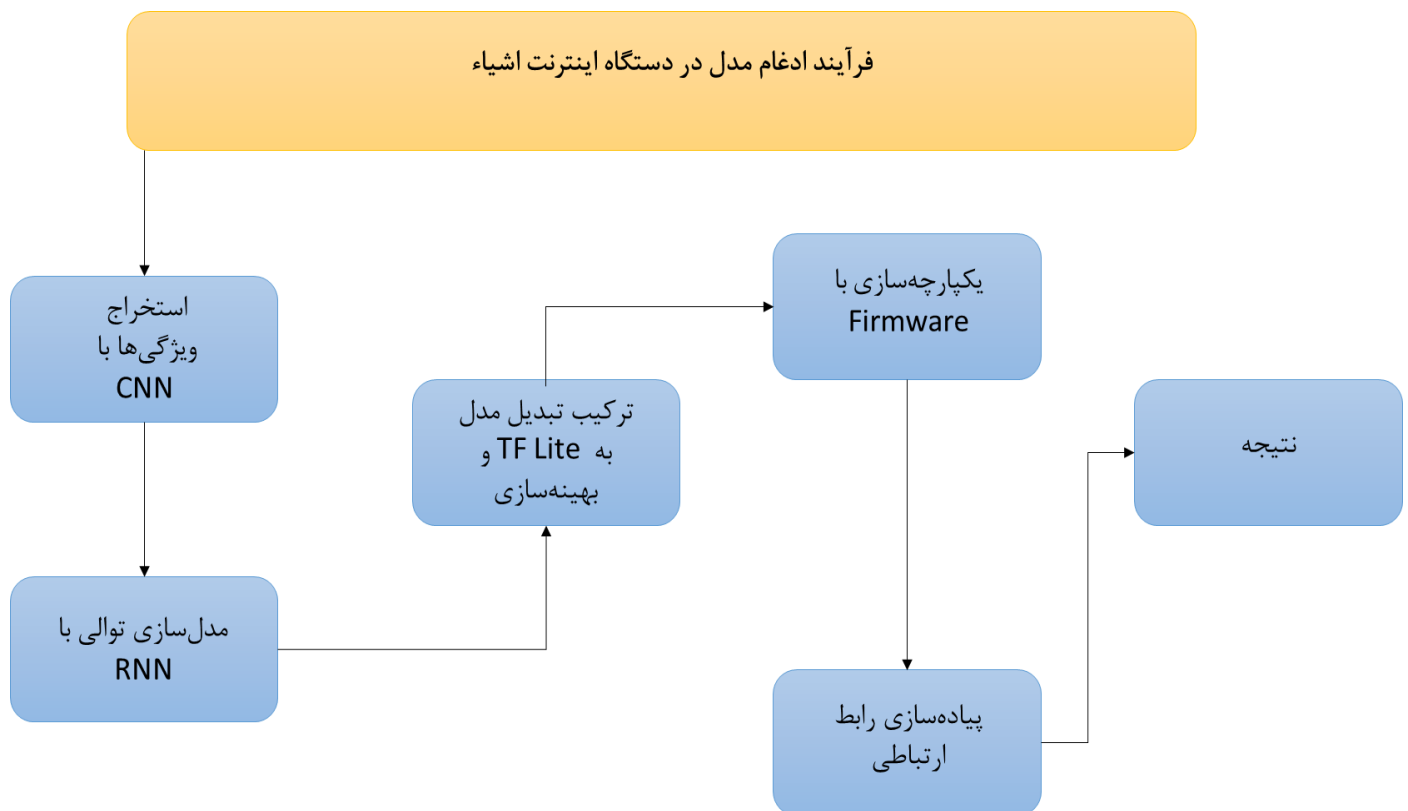
آزمایش در محیط واقعی

در نهایت، سیستم در محیط صنعتی واقعی مورد آزمایش قرار می گیرد تا عملکرد آن در شرایط عملیاتی واقعی ارزیابی شود. این مرحله برای تأیید صحت مدل در شرایط بدون شبیه سازی و بررسی اثربخشی آن در شناسایی ناهنجاری ها و پیش بینی مشکلات به کار می رود.

نتیجه گیری از آزمایش ها:

آزمایش ها در محیط های کنترل شده، شبیه سازی شده و واقعی نشان دادند که سیستم هوش مصنوعی با استفاده از مدل های ترکیبی CNN-RNN توانایی شناسایی ناهنجاری ها را با دقت بالا در بلادرنگ دارد. این آزمایش ها همچنین کمک کردند تا مشکلات احتمالی سیستم شناسایی شوند و بهینه سازی های بیشتری انجام گردد تا سیستم در شرایط مختلف و در زمان واقعی به طور مؤثر عمل کند.

این فرآیند اعتبارسنجی و آزمایش کمک می کند تا اطمینان حاصل شود که سیستم به طور مؤثر و با دقت بالا در محیط های صنعتی مختلف اجرا می شود و به طور مؤثری ناهنجاری ها و مشکلات احتمالی را پیش بینی و مدیریت می کند.



شکل ۳: فرآیند ادغام مدل در دستگاه اینترنت اشیا

کنترل برخی متغیرها:

این نوع آزمایش، دیدگاه واقع بینانه تری از عملکرد سیستم ارائه می دهد و به شناسایی مشکلاتی که ممکن است در محیط کنترل شده آشکار نشوند، کمک می کند. به عنوان مثال، یک سناریو می تواند 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 می شود که اطلاعات ارزشمندی درباره جنبه های مختلف سیستم فراهم می کنند. این معیارها به ما کمک می کنند تا کارایی سیستم را در شرایط واقعی و عملیاتی سنجیده و بهینه سازی های لازم را اعمال کنیم:

Accuracy

این معیار نسبت پیش بینی های صحیح مدل به کل پیش بینی ها را اندازه گیری می کند 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 دستگاه های IoT با منابع محدود، زمان تأخیر (زمان پردازش یک ورودی و تولید خروجی) اهمیت زیادی دارد. این معیار بر حسب میلی ثانیه (ms) اندازه گیری می شود؛ هرچه کمتر باشد، سیستم کاراتر است [14].

Resource

Usage

اطلاع از میزان استفاده از منابع مانند CPU، حافظه و انرژی حیاتی است، به ویژه در دستگاه های دارای منابع محدود. این معیار می تواند به صورت درصد یا مقدار مطلق بیان شود.

Recall

نسبت موارد مثبت واقعی که به درستی شناسایی شده اند. 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 شناسایی شده به کل موارد مثبت پیش بینی شده را اندازه گیری می کند [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 کلاس ها نامتعادل هستند، مفید است. به فرمول (4) دقت کنید.

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

اهمیت

معیارها:

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 نشان دهنده 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 ثبت شده توسط حسگر رطوبت در طول ۱۰ ساعت را نشان می‌دهد. این اندازه‌گیری‌ها توانایی سیستم در شناسایی و مستندسازی تغییرات کوچک در رطوبت محیط کنترل شده را منعکس می‌کند. سطح رطوبت بین ۴۴٪ و ۷۰٪ تغییر کرده است. به‌ویژه در سه زمان خاص (ساعت‌های ۵، ۶ و ۸)، حد تعیین شده قبلی تجاوز کرده و هشدارها فعال شده است. این حوادث تجاوز از حد نشان دهنده 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 نشان دهنده 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 ثبت شده توسط حسگر فشار را در طول ۱۰ ساعت نشان می‌دهد. این اندازه‌گیری‌ها توانایی سیستم در شناسایی و ثبت تغییرات فشار در محیط کنترل شده 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 تعیین شده در سه زمان خاص (ساعت های ۶، ۷ و ۹) عبور کرده و هشدارها فعال شده است. این هشدارها دقت سیستم را با شناسایی تغییرات قابل توجه در فشار نشان می دهند.

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 می کند. این ویژگی نشان می دهد که سیستم در برابر تغییرات مقاوم است و ابزاری حیاتی برای تضمین ایمنی و کارایی تجهیزات در حال کار تحت سطوح خاص فشار، مانند دیگ های بخار یا سیستم های هیدرولیک، محسوب می شود.

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 میزان های ثبت شده توسط شتاب سنج در طول ۱۰ ساعت را نشان می دهد. این داده ها ارتعاشاتی را که در محیط کنترل شده شناسایی شده است، منعطف می کند که به صورت متر بر ثانیه مربع (m/s^2) بیان می شود. ارتعاشات در ماشین آلات صنعتی یک شاخص حیاتی برای وضعیت و عملکرد آن ها به شمار می روند. افزایش غیرعادی در این ارتعاشات ممکن است نشان دهنده سایش زود هنگام یا مشکلی در عملکرد ماشین باشد. در طول دوره ثبت شده، ارتعاشات بین m/s^2 ۰,۰۰۲ و m/s^2 ۰,۰۰۸ متغیر بوده اند. در ساعت های ۲، ۵ و ۹، در سه زمان خاص، ارتعاشات از حد تعیین شده قبلی فراتر رفته و سیستم هشدارهایی را ایجاد کرده است. این اندازه گیری ها و هشدارها توانایی سیستم را در شناسایی تغییرات ظریف اما قابل توجه در ارتعاشات نشان می دهند و دقت بالایی را به نمایش می گذارند. بنابراین، این ابزار به عنوان ابزاری ضروری در شناسایی زود هنگام مشکلات یا خرابی های احتمالی در تجهیزات صنعتی معرفی می شود. جدول ۷ مشاهدات دوربین IoT را در طول ۱۰ ساعت جزئیات می دهد. این جدول مقایسه ای بین خطاهای عمدی وارد شده به سیستم و خطاهایی که دوربین شناسایی کرده، ارائه می دهد. مشاهده می شود که هم زمانی بالایی بین خطاهای وارد شده و شناسایی شده توسط دوربین در طول ساعات ثبت شده وجود دارد. با این حال، در زمان های خاص، مانند ساعت دوم و هشتم، تفاوت کمی بین خطاهای وارد شده و آن هایی که دوربین شناسایی کرده است، مشاهده می شود. در مجموع، ۲۳ خطا وارد شد و دوربین به طور دقیق ۲۱ مورد را شناسایی کرد که این معادل یک نرخ موفقیت ۹۱,۳٪ است. این داده ها 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 داده های ارتعاش و شتاب سنج استفاده می کند. در ادامه روش کار و گام های اصلی این مدل برای رسم نمودار بهبود یافته تشریح شده است:

دریافت داده های خام شتاب سنج

داده ها شامل سیگنال های ارتعاش از شتاب سنج، ثبت شده در هر دو محیط کنترل شده و شبیه سازی شده واقعی هستند.

این داده ها معمولاً در قالب سری های زمانی ثبت می شوند.

استفاده از CNN برای استخراج ویژگی ها

شبکه عصبی CNN (شبکه عصبی کانولوشنی) مسئول استخراج ویژگی های مکانی و محلی از سیگنال های ارتعاش است.

با استفاده از لایه های کانولوشن، الگوهای تکرار شونده و تغییرات جزئی در داده ها شناسایی می شوند.

این کار باعث می شود ویژگی های کلیدی و مهم از داده های خام شتاب سنج استخراج شوند.

استفاده از RNN برای تحلیل وابستگی های زمانی

پس از استخراج ویژگی ها توسط CNN، این ویژگی ها به شبکه RNN (شبکه بازگشتی) انتقال داده می شوند.

RNN توانایی تحلیل وابستگی های زمانی در داده ها را دارد و برای سیگنال های سری زمانی بسیار مؤثر عمل می کند.

با استفاده از واحدهایی مثل LSTM یا GRU، این شبکه می تواند تغییرات زمانی و رفتارهای غیرخطی ارتعاشات را پیش بینی و تشخیص دهد.

RNN کمک می کند که ناهنجاری ها و الگوهای پیچیده خطا شناسایی شوند.

ترکیب خروجی و صدور هشدار نهایی

خروجی ترکیبی از دو شبکه ویژگی های CNN و تحلیل زمانی RNN به عنوان ورودی به یک لایه تصمیم گیری منتقل می شود.

این لایه به کمک آستانه های یادگیری شده تصمیم می گیرد 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 می شوند.

برتری های مدل ترکیبی CNN-RNN نسبت به روش DNN:

افزایش دقت تشخیص خطا

در مدل DNN، شتاب سنج به دلیل نوسانات محیطی گاهی هشدارهای کاذب یا ناقص صادر می کند.

مدل CNN-RNN با استخراج ویژگی های عمیق و تحلیل وابستگی های زمانی می تواند به دقت بالاتری در تشخیص خطاها برسد.

قابلیت تطبیق پذیری با محیط های غیر کنترل شده

روش DNN در محیط های شبیه سازی شده و غیر خطی با خطا روبه رو می شود.

مدل CNN-RNN با قدرت تحلیل الگوهای پیچیده، می تواند در محیط های متغیر عملکرد بهتری داشته باشد.

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

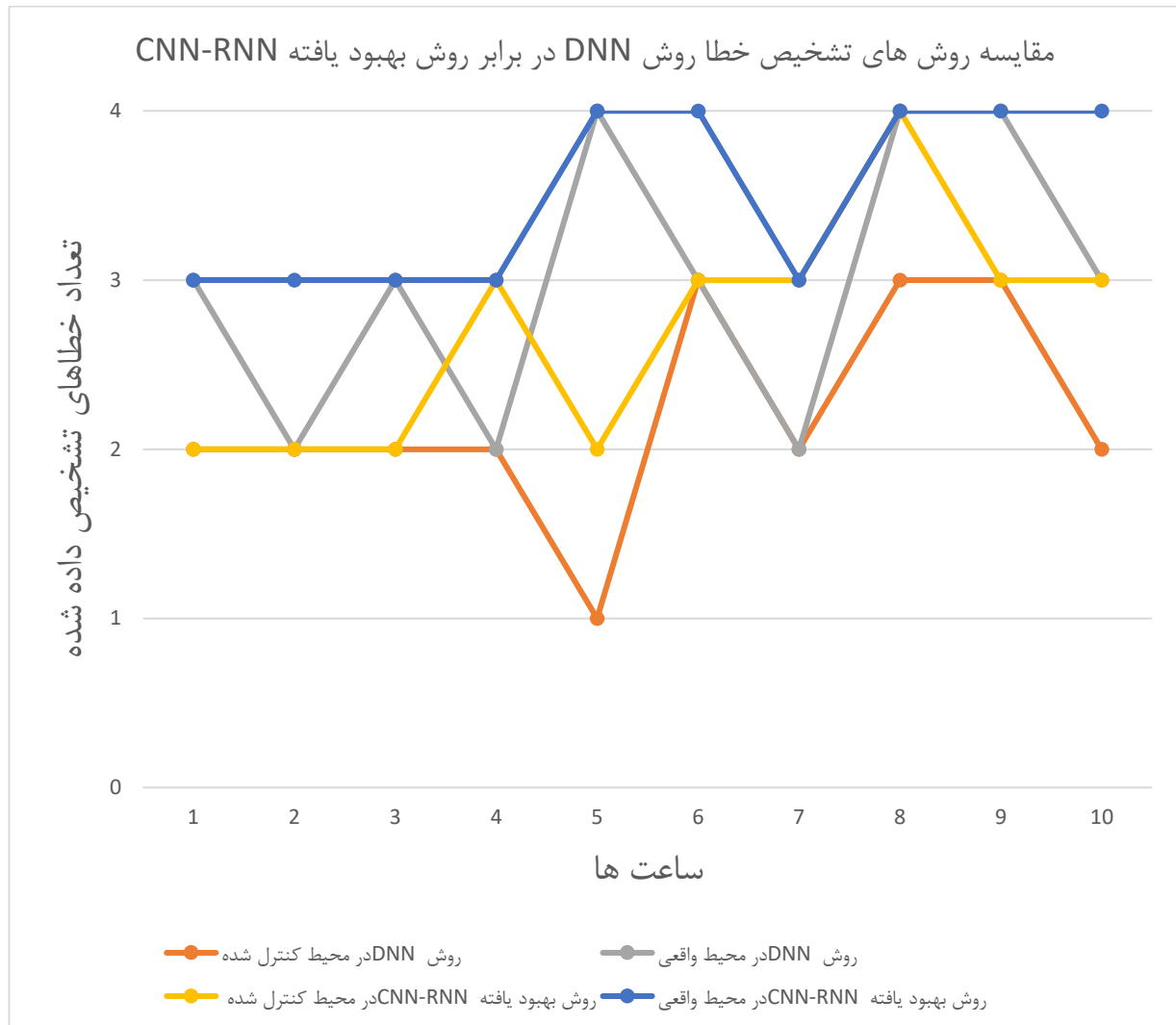
قابلیت یادگیری و به روز رسانی: مدل CNN-RNN بر اساس داده های جدید می تواند یادگیری مداوم داشته باشد و با گذر زمان بهبود یابد.

تحلیل رفتارهای غیر خطی: در شرایط واقعی، ارتعاشات و تغییرات شتاب پیچیده و غیر خطی هستند که مدل CNN-RNN به خوبی می تواند آن ها را تحلیل کند.

افزایش سرعت و دقت در پردازش داده های حجیم

ترکیب CNN و RNN به سیستم این امکان را می دهد که حجم زیادی از داده های سری زمانی را به سرعت پردازش و تحلیل کند.

مدل ترکیبی CNN-RNN به دلیل توانایی در استخراج ویژگی های عمیق با CNN و تحلیل وابستگی های زمانی با RNN، عملکردی به مراتب بهتر نسبت به روش سنتی شتاب سنج دارد. در نتیجه نمودار بهبود یافته نشان می دهد که تعداد خطاهای تشخیص داده شده در محیط های شبیه سازی شده واقعی بهبود یافته و هشدارهای کاذب نیز کاهش چشمگیری پیدا کرده اند.



شکل 4: مقایسه روش های تشخیص خطا روش 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 مقایسه ای میان روش بهبود یافته CNN-RNN در دو محیط کنترل شده و واقعی و مقایسه آن با روش PdM

۱- داده محور بودن رویکرد نگهداری پیش بینانه (PdM)

- تعریف PdM: این رویکرد از داده های بلند مدت و تاریخی استفاده می کند تا پیش از وقوع خرابی تجهیزات، آن ها را شناسایی و پیش بینی کند.
- داده ها از طریق سنسورهای متنوعی مانند سنسورهای لرزش، دما، فشار، و مصرف برق جمع آوری می شوند.

۲- نقش هوش مصنوعی (AI) در PdM

- یادگیری ماشین (ML): الگوریتم های نظارت شده و غیر نظارتی مانند SVM، درخت تصمیم گیری، و خوشه بندی k-Means برای شناسایی ناهنجاری ها و پیش بینی سلامت تجهیزات استفاده می شوند.
- یادگیری عمیق (DL): استفاده از CNN برای استخراج ویژگی ها از داده های پیچیده و RNN به ویژه LSTM برای درک وابستگی های زمانی و پیش بینی عمر باقی مانده تجهیزات (RUL).

۳- استفاده از هم افزایی IIoT و AI

- اینترنت اشیای صنعتی: سیستم های IIoT داده ها را از تجهیزات صنعتی جمع آوری و انتقال می دهند.
- پردازش و تحلیل داده های حجیم به کمک مدل های AI برای شناسایی الگوهای پیچیده و استخراج بینش های عملیاتی.

۴- نقش هم جوشی داده ها: (Sensor Fusion)

- داده های جمع آوری شده از چند سنسور ترکیب می شوند تا تصویر جامع تری از سلامت تجهیزات ایجاد کنند.

- این فرآیند باعث افزایش دقت شناسایی ناهنجاری ها و کاهش خطاهای مثبت کاذب می شود.

۵- چارچوب کلی Pdm بلادرنگ:

۱. جمع آوری داده: سنسورها اطلاعات بلادرنگ را ثبت می کنند.
۲. پیش پردازش داده: شامل حذف نویز و نرمال سازی داده ها.
۳. مدل سازی هوش مصنوعی: استفاده از DL و ML برای شناسایی ناهنجاری ها و پیش بینی خرابی ها.
۴. تصویری سازی داده ها: ارائه اطلاعات پردازش شده به کاربران از طریق رابط کاربری.
۵. تصمیم گیری: ارائه پیشنهادات برای اقدامات تعمیر و نگهداری به موقع.

۶- مزایای روش:

۱. کاهش خرابی های برنامه ریزی نشده.
۲. بهبود بهره وری عملیاتی و کاهش هزینه ها.
۳. تصمیم گیری مبتنی بر داده برای بهینه سازی منابع.
۴. افزایش ایمنی کارکنان از طریق شناسایی زود هنگام خطرات.

این روش با ترکیب قدرت هوش مصنوعی و IIoT، امکان دستیابی به نگهداری پیش بینانه مؤثر و کاهش هزینه های عملیاتی را فراهم می کند.

در جدول ۸ روش Pdm نشان دهنده عملکردی متوسط با نوسانات بیشتر نسبت به سایر روش ها است.

روش CNN-RNN در محیط کنترل شده عملکرد بهتر و پایداری نسبت به Pdm دارد.

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

شکل ۴ به طور گرافیکی پیشرفت و بهبود عملکرد مدل را با گنجانیدن روش ترکیبی نشان می دهد. این شکل، تکامل عملکرد مدل با و بدون بازخورد و همچنین روش CNN-RNN را طی ۱۲ هفته به تصویر می کشد. دقت اولیه مدل که نشان دهنده 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 ارائه شده بهره مند می شود و دقت آن در هفته دوازدهم تقریباً به ۹۶.۲٪ می رسد. اما در روش CNN-RNN دقت در هفته اول از ۹۲.۵٪ به ۹۹.۶۵٪ در هفته دوازدهم رسیده که بسیار چشمگیر می باشد.

روش CNN-RNN با سرعت بیشتری بهبود می یابد و دقت آن از ۹۲.۵ درصد در ساعت اول شروع شده و در ساعت دوازدهم به ۹۹.۶۵ درصد می رسد. این رشد سریع نشان دهنده کارایی بالای این روش در یادگیری و پیش بینی است. در مقابل، روش ترکیبی از الگوریتم ها با دقت اولیه ۹۰ درصد شروع می شود و به تدریج در ساعت دوازدهم به دقت ۹۷ درصد می رسد. این رشد کندتر نشان دهنده نیاز این روش به زمان بیشتر یا تنظیمات بهینه تر برای رسیدن به دقت بالاتر است. مقایسه این روش ها نشان می دهد که CNN-RNN برای وظایفی که نیاز به یادگیری سریع تر و دقت بالاتر دارند، مناسب تر است.

۱. روش CNN-RNN

برتری:

ترکیب CNN و RNN به طور خاص برای داده های پیچیده و زمانی طراحی شده است. در این نمودار، ترکیب CNN برای استخراج ویژگی های مکانی (مانند داده های تصویری یا ویژگی های حسگر) و RNN برای تحلیل وابستگی های زمانی (مانند پیش بینی در طول زمان) استفاده شده است.

دقت بالا: در مقایسه با سایر روش ها، CNN-RNN بالاترین دقت را نشان می دهد، به ویژه با گذشت زمان. این نشان می دهد که این ترکیب به طور مؤثر وابستگی های زمانی و مکانی را در داده ها شبیه سازی می کند و توانایی آن در شناسایی الگوهای پیچیده در داده های زمان بندی شده بسیار بالاست.

مزیت اصلی: توانایی این روش در به دست آوردن دقت بالاتر با افزایش زمان است، که به دلیل یادگیری بهتر وابستگی های بلندمدت در داده ها است.

علت برتری:

توانایی یادگیری وابستگی های بلندمدت: استفاده از LSTM در مدل RNN باعث می شود این روش بتواند الگوهای زمانی بلندمدت را یاد بگیرد و از اطلاعات قبلی استفاده کند، که برای داده های IoT که نیاز به پیش بینی در طول زمان دارند بسیار مؤثر است.

ترکیب ویژگی های مکانی و زمانی: مدل CNN-RNN این قابلیت را دارد که ویژگی های مکانی را از داده های تصویری استخراج کرده و سپس وابستگی های زمانی را پردازش کند، که این ویژگی در بسیاری از داده های واقعی به ویژه داده های IoT بسیار مفید است.

۲. روش DNN شبکه های عصبی عمیق

ضعف:

کاهش دقت: در مقایسه با CNN-RNN، DNN دقت کمتری در پردازش داده ها به ویژه در زمان های طولانی تر نشان می دهد.

عدم پردازش وابستگی های زمانی: به طور کلی برای تحلیل داده های زمانی طراحی نشده است و این موضوع باعث کاهش دقت در پیش بینی ها می شود، به ویژه زمانی که داده های وابسته به زمان (سری زمانی) هستند.

ناتوانی در یادگیری الگوهای پیچیده زمانی: DNN تنها از ساختار لایه ای و ویژگی های اولیه برای پیش بینی استفاده می کند، در حالی که نمی تواند وابستگی های زمانی طولانی مدت را به طور مؤثر شبیه سازی کند.

علت ضعف:

کمبود تحلیل وابستگی های زمانی: DNN از لحاظ ساختاری تنها ویژگی ها را پردازش می کند و به طور مستقیم از داده های سری زمانی استفاده نمی کند. بنابراین، در مواجهه با داده هایی که به وابستگی های زمانی نیاز دارند، عملکرد آن ضعیف است.

۳. روش های دیگر مثل SVM و Random Forest

ضعف:

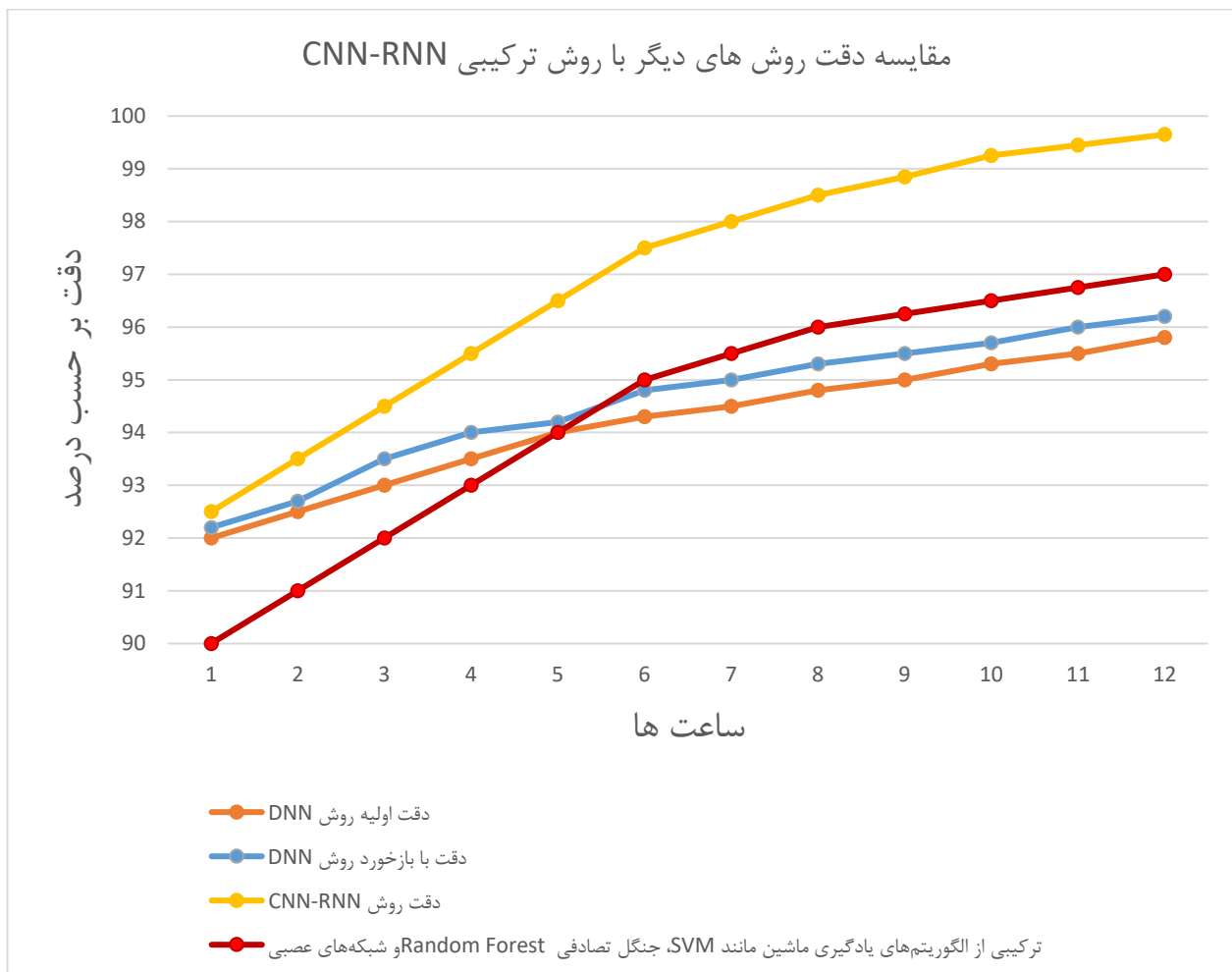
دقت پایین تر: این روش ها مانند SVM و Random Forest نسبت به CNN-RNN دقت پایین تری دارند، به ویژه در زمان هایی که نیاز به پردازش داده های وابسته به زمان است.

کمتر تطبیق پذیر: این مدل ها برای داده های سری زمانی طراحی نشده اند و بنابراین قادر به یادگیری وابستگی های بلندمدت نیستند.

علت ضعف:

عدم توانایی در یادگیری وابستگی های زمانی: روش هایی مانند SVM و Random Forest معمولاً برای داده های ایستا و مستقل از زمان طراحی شده اند. بنابراین، در پردازش داده های زمانی که به یادگیری وابستگی های پیچیده نیاز دارند، این روش ها ضعف دارند.

عدم پردازش داده های پیچیده: در حالی که این مدل ها می توانند در مسائل غیر زمانی عملکرد خوبی داشته باشند، اما برای مسائل پیچیده تر که نیاز به پردازش همزمان ویژگی های 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 شکست ها یا روندهای غیرعادی، تصمیم گیری ها دیگر تنها در پاسخ به مشکلات پس از بروز آنها اتخاذ نمی شود. در عوض، تصمیمات به طور پیش بینانه و با چشم انداز اتخاذ می شوند، که از شکست ها یا وقفه های پرهزینه جلوگیری می کند. این تغییر به سمت تصمیم گیری آگاهانه و پیش دستانه، برای بهینه سازی بهره وری و کاهش هزینه های 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. RNN 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 Gummadi, 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]