

## مقایسه دو روش شبکه عصبی مصنوعی عمیق و شبکه عصبی پیچشی در

## طبقه‌بندی عیوب جعبه‌دنده

مصطفی عابدی  
استادیار  
دانشگاه شهید بهشتی  
mo\_abedi@sbu.com

عباس روحانی بسطامی\*  
استادیار  
دانشگاه شهید بهشتی  
A\_rohani@sbu.com

آرمین فهندژ  
کارشناسی ارشد  
دانشگاه شهید بهشتی  
fahandezharmin@gmail.com

تاریخ پذیرش: ۱۴۰۲/۱۱/۲۹

تاریخ دریافت: ۱۴۰۲/۰۷/۰۴

## چکیده

امروزه نظارت بر وضعیت ماشین‌آلات و تشخیص هوشمند عیوب برای تولیدات صنعتی نقش بسیار پراهمیتی را داراست. روش‌های هوش مصنوعی برای پایش در مقیاس‌های بزرگ (کلان‌داده<sup>۱</sup>) بدون هیچ‌گونه فرض آماری در مورد داده‌ها می‌توانند به‌درستی عمل کنند. در این پژوهش مقایسه دو روش شبکه عصبی مصنوعی عمیق و شبکه عصبی پیچشی<sup>۲</sup> در طبقه‌بندی عیوب جعبه‌دنده انجام شده است. در روش شبکه عصبی مصنوعی عمیق ویژگی‌ها از سیگنال زمانی شتاب استخراج شده و در روش دیگر از خود سیگنال به‌عنوان ورودی استفاده شده است. به‌طور خلاصه از این روش‌ها برای طبقه‌بندی ۹ حالت معیوب و یک حالت سالم در ۶ ترکیب سرعت و بار متفاوت استفاده و با یکدیگر مقایسه شده است و سپس به بیان اثر چالش‌هایی از قبیل طول پنجره، ضریب یادگیری و تعیین تعداد ویژگی‌ها و چگونگی برطرف کردن آنها پرداخته شده است. در انتها با قیاس نتایج به‌دست آمده از هر دو روش این نتیجه حاصل شد که قدرت تشخیص شبکه عصبی پیچشی در این مورد بهتر از روش دیگر است.

واژگان کلیدی: شبکه عصبی پیچشی، یادگیری عمیق، ارتعاش، جعبه‌دنده، عیب‌یابی

## ۱. مقدمه

دوره‌های تعمیرات اساسی، ممکن است قطعاتی تعویض شوند که هنوز کارایی خود را دارند. امروزه با پیشرفت علم و فناوری در انواع صنایع تغییرات عظیمی هم در حیطه پایش تجهیزات ایجاد شده است [۲، ۱]، که از میان آنها می‌توان به کلان‌داده اشاره کرد که در ایجاد نوآوری در روش‌های عیب‌یابی بسیار مؤثر است [۳]. با توجه به روند رشد سریع اطلاعات در اثر پیشرفت فناوری، داده‌هایی در حجم وسیع ایجاد شد که فراتر

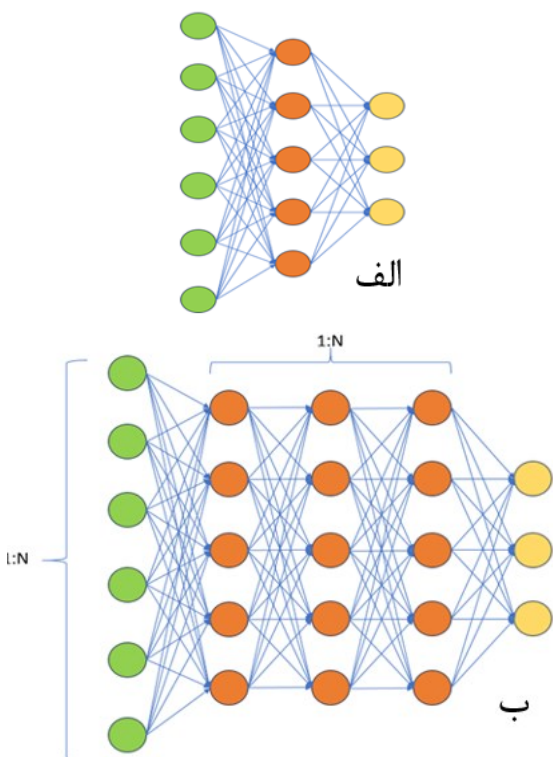
در گذشته، شناسایی عیوب تجهیزات با بازدیدهای میدانی انجام می‌شد. اما این روش دارای مشکلاتی مانند دوری و عدم دسترسی به تجهیزات، کمبود نیروی ماهر جهت بازدید، تشخیص اشتباه و یا عدم تشخیص بعضی از عیوب توسط نیروی انسانی است. بازدیدها معمولاً به‌صورت دوره‌ای صورت می‌گیرد و ممکن است مشکلی میان دو دوره اتفاق افتد. همچنین به‌علت عدم‌توانایی در شناخت برخی عیوب، در

از توانایی انسان در بررسی آنها است. به همین منظور بیش از گذشته، نیاز به یک سیستم هوشمند برای رصد دائم این اطلاعات و همچنین استفاده از الگوریتم‌ها برای پیش‌بینی وضعیت سیستم احساس می‌شود. بدیهی است که تشخیص به‌موقع و درست عیوب در پایش وضعیت تجهیزات نقش مهمی در سلامت آن دارد [۴]. با توجه به روش‌های مختلف موجود در عیب‌یابی خودکار، در صنعت بیشتر از الگوریتم‌های مربوط به روش‌های کلاسیک [۵، ۶] و یادگیری ماشین که زیرمجموعه اصلی آن شبکه‌های عصبی است، استفاده می‌شود [۷، ۸]. براساس بررسی‌ها و تحقیقات انجام شده می‌توان پیشرفت شبکه‌های عصبی و یادگیری عمیق را در سال‌های اخیر مشاهده کرد، که خود نشان از استفاده روز افزون این نوع الگوریتم‌ها برای تشخیص و پیش‌بینی عیوب است [۹].

جعبه‌دنده یک سیستم انتقال قدرت است که با استفاده از مجموعه‌های مختلف چرخ‌دنده، می‌تواند سرعت یا گشتاور یک منبع قدرت را تغییر دهد و در ۲ دسته کاهنده (کاهش سرعت خروجی) و افزایش‌دهنده (افزایش سرعت خروجی) تقسیم‌بندی می‌شود. بر روی جعبه‌دنده امکان رخ دادن عیوبی اعم از سایش، شکستگی، لب‌پریدگی دندانه، نابالانسی و... وجود دارد که به‌علت بسته بودن این سیستم امکان تشخیص عیب وجود ندارد و برای تشخیص آن باید کل سیستم متوقف شده و جعبه‌دنده باز شود که این خود هزینه‌های زیادی را به‌بار می‌آورد. پس نیاز است که روشی ایجاد شود که بدون تحمیل هزینه‌های گزاف فرایند عیب‌یابی لحاظ شود. جعبه‌دنده‌ها در بسیاری از صنایع کاربرد دارند و در انواع مختلفی مانند خورشیدی، حلزونی و سایر مدل‌ها تولید می‌شوند.

شبکه‌های عصبی، یک روش محاسباتی براساس داده‌های ورودی و پیش‌بینی پاسخ‌های خروجی از سامانه‌های پیچیده هستند. ایده اصلی اینگونه شبکه‌ها تا حدودی الهام گرفته از شیوه کارکرد سیستم عصبی زیستی برای پردازش داده‌ها و اطلاعات به‌منظور یادگیری است. به‌طور کلی شبکه‌های عصبی

به دو دسته عمیق و کم‌عمق تقسیم‌بندی شده‌اند. ساختمان این شبکه از سه بخش اصلی تشکیل شده که عبارت است از لایه ورودی که مربوط به داده‌هایی است که به شبکه تحویل می‌شود، لایه‌های پنهان که برای تحلیل و یادگیری استفاده می‌شود و لایه خروجی که برای طبقه‌بندی و نمایش خروجی شبکه استفاده می‌شود. در شکل ۱ تفاوت دو نوع شبکه عصبی نمایش داده شده است.



شکل ۱. شبکه عصبی کم‌عمق (الف) شبکه عصبی عمیق (ب) [۱۰]

یادگیری عمیق به‌عبارت دیگر همان شبکه عصبی است که از تعداد بیشتری لایه مخفی نسبت به شبکه عصبی کم‌عمق برای محاسبات استفاده می‌کند [۱۰]. در پایش وضعیت هوشمند به‌وسیله شبکه عصبی، ابتدا چند نمونه از داده‌های تجهیز جمع‌آوری و به لایه ورودی شبکه داده می‌شوند. در ادامه شبکه عصبی با استفاده از الگوریتم‌های یادگیری، از داده‌های آموزشی الگوهایی را در سیستم تشخیص می‌دهد [۱۱].

در یک بررسی پرایوین کومار و همکاران جعبه‌دنده مربوط به یک خودرو را عیب‌یابی کرده و توسط روش ماشین بردار پشتیبان بررسی کردند. در این جعبه‌دنده هر یک از ۴ چرخ‌دنده موجود را یک بار در حالت سالم و یک بار هم در حالت معیوب داده‌برداری کردند. همچنین هر حالت با دو سرعت (500rpm و 750) و دو بار متفاوت (N.m0 و 5) آزمایش شده است. نتیجه حاصل از آزمایش نشان داد که دقت تشخیص این روش برای هر چهار چرخ‌دنده بالای 90 درصد است [12].

اوژان و همکاران عیب‌یابی را بر روی یک جعبه‌دنده تک مرحله انجام دادند. تجهیز آنها از یک پینیون 19 دندانه و یک چرخ‌دنده 48 دندانه تشکیل شده است. داده‌برداری در 4 مرحله صورت گرفت، یک بار در حالت سالم و 3 بار در حالت معیوب، که خرابی بر روی پینیون ایجاد شده است. هدف آنها بررسی و مقایسه پاسخ‌های به‌دست آمده با استفاده از 3 روش سنتی یادگیری ماشین (بردار پشتیبان<sup>3</sup>، درخت تصمیم<sup>4</sup>، جنگل تصادفی<sup>5</sup>) و یک روش یادگیری عمیق است. در نهایت مشخص شد که روش یادگیری عمیق دقت بالاتری دارد [13]. در آزمایش دیگری سامانتا و همکاران برای عیب‌یابی یک یاتاقان از دو الگوریتم شبکه عصبی مصنوعی عمیق و ماشین بردار پشتیبان استفاده کرده و بهینه‌سازی آن با استفاده از الگوریتم ژنتیک انجام داده‌اند. از مقایسه الگوریتم‌ها نتایج به‌دست آمده نشان داد عملکرد ماشین بردار پشتیبان از شبکه عصبی مصنوعی عمیق در این مورد به‌طور کلی بهتر بوده و همچنین زمان آموزش ماشین بردار پشتیبان کمتر از دیگری بوده است [14].

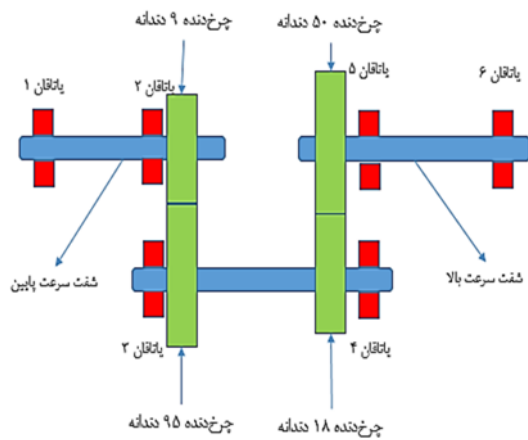
در بررسی دیگری ژئی جنگ چن و همکاران عیب‌یابی ترکیبی را بر روی یک جعبه‌دنده انجام دادند. این جعبه‌دنده از 4 چرخ‌دنده و 6 یاتاقان تشکیل شده است. در این پژوهش 12 شرط الگویی مختلف از خرابی لحاظ شده

و در 5 دور و 4 بار متفاوت داده‌برداری انجام شد که به‌عبارت دیگر از هر مورد شرایط الگویی 20 بار داده‌برداری شده است. بررسی و مقایسه داده‌ها با استفاده از دو روش شبکه عصبی پیچشی و ماشین بردار پشتیبان صورت گرفته است و در نتیجه بالاترین دقت شبکه عصبی پیچشی برابر با 98/9 درصد و در شبکه داده‌بردار پشتیبان برابر با 81/5 درصد بود. اما میانگین دقت در شبکه عصبی پیچشی در 20 تکرار برابر با 96/8 درصد و در روش دیگر برابر با 67/8 درصد است [15].

در این پژوهش عیب‌یابی بر روی یک جعبه‌دنده صورت می‌گیرد تا بتوان انواع خرابی ایجاد شده بر روی آن را از یکدیگر و همچنین از حالت سالم تشخیص داد. برای انجام این کار از روی جعبه‌دنده داده‌های شتاب در حوزه زمان در 2 جهت عمود بر هم یعنی در 2 دسته داده‌برداری شد، سپس داده‌های خام از هر دسته پیش پردازش و با یکدیگر ترکیب شده است. در ادامه از دو روش شبکه عصبی مصنوعی عمیق و شبکه عصبی پیچشی که در سال‌های اخیر بسیار مورد استفاده محققین بوده برای تفکیک حالات مختلف خرابی استفاده شد و در انتها دقت تشخیص شبکه‌ها با یکدیگر مقایسه شده است.

## 2. چیدمان آزمایشگاهی

دستگاه مورد آزمایش جعبه‌دنده اصلی یک توربین باد شبیه‌سازی شده است که داده‌برداری و عیب‌یابی بر روی آن انجام می‌گیرد. پره این توربین توسط یک موتور الکتریکی به حرکت در می‌آید و انرژی تولیدی حاصل از ژنراتور آن تعداد مشخصی از لامپ‌ها را می‌تواند روشن نماید. اجزای این جعبه‌دنده افزایش یافته که سرعت rpm 40 شفت ورودی را به rpm 1170 شفت خروجی تبدیل می‌کند، شامل 4 چرخ‌دنده، 6 یاتاقان، 3 شفت داخلی و پوسته است که در شکل 2 تصویر کلی دستگاه و در شکل 3 نمای داخل جعبه‌دنده نشان داده شده است.



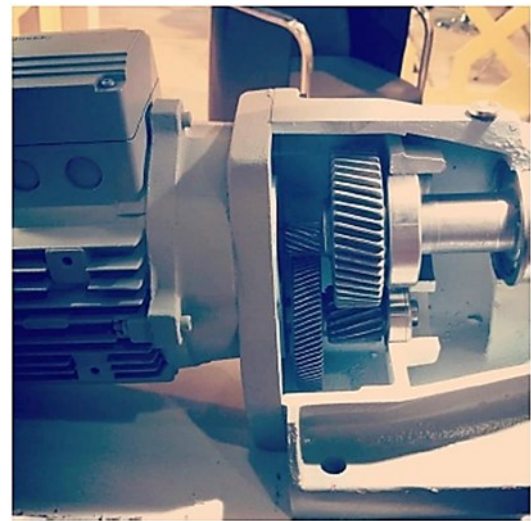
شکل ۴. نحوه قرارگیری اجزا جعبه‌دنده



شکل ۲. تصویر کلی دستگاه [۲۶]

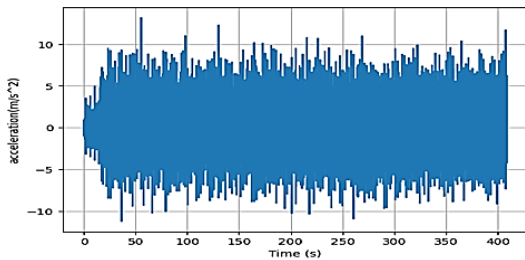
داده‌برداری در ۱۰ کلاس مختلف صورت گرفت که یک کلاس آن سالم و در سایر کلاس‌ها انواع خرابی به شرح زیر آمده است.

- ۱- Normal (سالم)
- ۲- Gear 18T wearing (سایش چرخ‌دنده ۱۸ دندانه)
- ۳- Gear 18T wearing & Bearing 2 (سایش چرخ‌دنده ۱۸ دندانه و بلبرینگ معیوب شماره ۲)
- ۴- Gear 18T wearing & Bearing 3 (سایش چرخ‌دنده ۱۸ دندانه و بلبرینگ معیوب شماره ۳)
- ۵- Gear 18T wearing & Bearing 4 (سایش چرخ‌دنده ۱۸ دندانه و بلبرینگ معیوب شماره ۴)
- ۶- Unbalance (نابالانسی)
- ۷- Gear 18T wearing & unbalance (سایش چرخ‌دنده ۱۸ دندانه و نابالانسی)
- ۸- Gear 95T Mashed (له‌شدگی چرخ‌دنده ۹۵ دندانه)
- ۹- Gear 95T Chipping (لب‌پریدگی چرخ‌دنده ۹۵ دندانه)
- ۱۰- Gear 95T Chipping & unbalance (لب‌پریدگی چرخ‌دنده ۹۵ دندانه و نابالانسی)

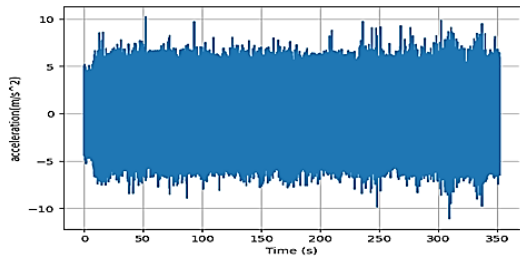


شکل ۳. نمای داخلی جعبه‌دنده [۲۶]

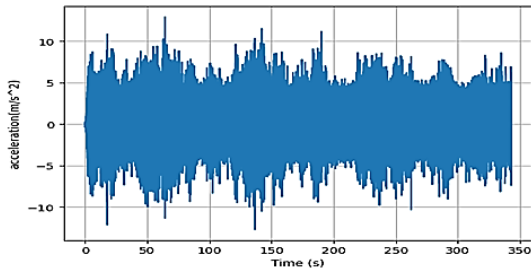
همان‌طور که در فصل قبل گفته شد عیوب مختلفی می‌تواند بر روی جعبه‌دنده ایجاد شود [۱۶] که در این پژوهش خرابی‌هایی که امکان ایجاد بود لحاظ شده است. داده‌برداری از جعبه‌دنده اصلی با استفاده از دو سنسور شتاب که بر یکدیگر عمود هستند و بر روی پوسته نصب شده‌اند صورت می‌گیرد. به‌طور کلی داده‌برداری در حالات، سرعت‌ها و بارهای مختلف انجام شد تا بتوان دقت تشخیص نهایی شبکه عصبی را به بالاترین حد ممکن رساند. در شکل ۴ نحوه قرارگیری اجزا جعبه‌دنده، موقعیت چرخ‌دنده‌ها و همچنین شماره‌گذاری یاتاقان‌ها نمایش داده شده است.



(الف)



(ب)



(ج)

شکل ۵. (الف) نمونه سیگنال نرمال از سیگنال‌های شتاب در حوزه زمان (ب) نمونه اول سیگنال معیوب از سیگنال‌های شتاب در حوزه زمان (ج) نمونه دوم سیگنال معیوب از سیگنال‌های شتاب در حوزه زمان

قابل ذکر است که ترکیب سرعت‌ها و بارهای مختلف نیز به صورت کلی در ۶ حالت داده‌برداری شده است.

۱. فرکانس ۲۰ هرتز و بار ۰ وات
۲. فرکانس ۳۰ هرتز و بار ۰ وات
۳. فرکانس ۴۲/۱ هرتز و بار ۰ وات
۴. فرکانس ۴۴/۵ هرتز و بار ۱۲۰ وات
۵. فرکانس ۴۶ هرتز و بار ۱۸۰ وات
۶. فرکانس ۴۹/۷ هرتز و بار ۳۰۰ وات

همان‌طور که مشخص است مجموعاً ۶۰ فایل داده‌برداری ایجاد شد که در هر کدام به صورت حدودی در بازه زمانی ۵ دقیقه با فرکانس نمونه‌برداری ۱۰ کیلوهرتز برداشت شده است. لازم به ذکر است که چون ۲ سنسور شتاب در جهات عمودی و افقی قرار دارد، برای هر حالت دو سری داده به دست می‌آید که در این پژوهش از هر دو آنها (دو کانال شدن ورودی شبکه) استفاده شده است. داده‌ها در هنگام روشن و خاموش شدن دستگاه هم ذخیره می‌شوند پس برای جلوگیری از بروز خطا در پروژه از داده‌های ۴ دقیقه میانی استفاده شد. زیرا در هنگام خاموش و روشن کردن دستگاه سرعت دچار نوسان شده و از فرکانس‌های طبیعی نیز عبور می‌کند که این خود خطایی در فرایند تحلیل ایجاد می‌کند. نمونه‌ای از سیگنال‌ها در شکل ۵ آمده است.

### ۳. یادگیری عمیق

یادگیری عمیق به دلیل توانایی در تشخیص الگوهای پیچیده و ارتباطات نهفته، از آنجایی که می‌تواند اطلاعاتی را که ابتدا به نظر پیچیده و درهم‌تنیده می‌آیند را به صورت ساده و قابل فهم برای سیستم‌های کامپیوتری تبدیل کند، در بسیاری از زمینه‌ها مورد استفاده قرار می‌گیرد [۱۷، ۱۸]. این روش در عیب‌یابی سیستم‌های دوار دارای چالش‌هایی مانند طول شبکه، دقت، زمان و ... است که در ادامه بررسی می‌شود [۱۹]. در این پروژه از دو نوع شبکه برای طبقه‌بندی و پیش‌بینی خرابی‌ها استفاده شده است. همچنین در این پژوهش از نرم‌افزار پایتون ۳ در بستر گوگل کولب<sup>۷</sup> استفاده شده است که CPU پیش‌فرض آن یک پردازنده Intel Xeon با ۲ vCPU (CPU مجازی) و ۱۳ گیگابایت RAM است و پردازنده‌های گرافیکی استاندارد آن NVIDIA T4 Tensor Core است.

#### ۳-۱. شبکه عصبی مصنوعی عمیق

در این روش از یک شبکه دارای لایه‌های متراکم استفاده شده است که لازم است ابتدا ویژگی‌های آماری از سیگنال‌های شتاب در حوزه زمان استخراج شود. ویژگی‌های در نظر گرفته شده مطابق با جدول ۱ است که در آن  $n$  برابر با تعداد کل مقادیر و  $\bar{x}$  برابر با میانگین داده‌ها است.

جدول ۱. ویژگی‌های آماری در حوزه زمان

$$X_{std} = \sqrt{\frac{\sum (x(i) - \bar{x})^2}{N}} \quad \text{Standard deviation} \quad (1)$$

$$X_{kur} = \frac{\left(\frac{1}{n} \sum (x(i) - \bar{x})^4\right)}{x_{std}^4} \quad \text{Kurtosis} \quad (2)$$

$$X_{ptp} = |\text{Max}(x(i))| + |\text{Min}(x(i))| \quad \text{Peak to peak} \quad (3)$$

$$X_{rms} = \sqrt{\frac{\sum (x(i))^2}{N}} \quad \text{Root Mean Square} \quad (4)$$

$$X_{mean} = \frac{\sum x(i)}{N} \quad \text{Mean} \quad (5)$$

$$X_{ske} = \frac{\left(\frac{1}{n} \sum (x(i) - \bar{x})^3\right)}{x_{std}^3} \quad \text{Skewness} \quad (6)$$

$$SF = \frac{x_{rms}}{\left(\frac{1}{n} \sum |x(i)|\right)} \quad \text{Shape factor} \quad (7)$$

$$CF = \frac{x_{peak}}{x_{rms}} \quad \text{Crest factor} \quad (8)$$

روش کار بدین صورت است که در هر حالت ۴ دقیقه‌ای داده‌برداری شد، پنجره‌ای به طول ۴۰۰۰ نقطه ایجاد کرده و سپس در طول سیگنال حرکت داده می‌شود. لازم به ذکر است که علت انتخاب این عدد به عنوان طول پنجره در بخش ۳-۳ بیان می‌شود. در هر پنجره ۸ ویژگی نام برده شده محاسبه و در ماتریسی ذخیره می‌گردد. ماتریس نهایی ایجاد شده به ابعاد (۱۰، ۷۲۰۰، ۸) که عدد ۱۰، برابر با تعداد کلاس‌ها و ۷۲۰۰\*۸ فضای مورد نیاز برای ذخیره ویژگی‌های محاسبه شده است. در مرحله بعد این ماتریس کلی برچسب‌گذاری شده و سپس به دو بخش آموزش و آزمون تقسیم‌بندی می‌شود که ۸۰ درصد داده‌های مربوط به آموزش و ۲۰ درصد دیگر مربوط به آزمون است. در قدم بعدی اطلاعات به شبکه داده شده و نتیجه

به دست می‌آید. در جدول ۲ ساختار شبکه عصبی مصنوعی عمیق آمده است. لازم به ذکر است که برای تعیین تعداد لایه‌ها از روش آزمون و خطا استفاده شده است بدین صورت که ابتدا شبکه با لایه‌ها و نورون‌های کمتر اجرا شده و با توجه به یادگیری و دقت شبکه به صورت صعودی به این لایه‌ها افزوده شده تا به بالاترین دقت ممکن برسد. تعداد نورون‌های لایه آخر به دلیل اینکه ۱۰ کلاس طبقه‌بندی داریم برابر با ۱۰ در نظر گرفته شده است.

جدول ۲. ساختار شبکه عصبی مصنوعی عمیق

```
Model: "sequential"
Layer (Dense (Output Shape Param #
=====
den (None, 32) 288
dense_1 (None, 64) 2112
dense_2 (None, 128) 8320
dense_3 (None, 256) 33024
dense_4 (None, 512) 131584
dense_5 (None, 256) 131328
dense_6 (None, 128) 32896
dense_7 (None, 64) 8256
dense_8 (None, 32) 2080
dense_9 (None, 10) 363
=====
Total params: 350,251
Trainable params: 350,251
Non-trainable params: 0
```

### ۳-۲. شبکه عصبی پیچشی

در این روش از شبکه عصبی پیچشی استفاده شده است. در این روش نیازی به استخراج ویژگی‌های آماری داده‌ها نیست زیرا شبکه قابلیت تشخیص الگو از روی داده‌های حوزه زمان را دارد [۲۱، ۲۰]. ابتدا پنجره‌ای به طول ۱۰۰۰ نقطه ایجاد کرده، زیرا بیشترین دقت در این طول ایجاد می‌شود و بر روی سیگنال شتاب حرکت داده می‌شود. علت انتخاب این عدد به عنوان طول پنجره در بخش ۳-۳ ذکر می‌شود. داده‌های

موجود در پنجره، در هر بار درون ماتریسی به ابعاد (۱۰، ۱۰، ۲۸۸۰۰، ۱۰۰۰) ذخیره می‌شود که عدد ۱۰، برابر با تعداد کلاس‌ها و ۲۸۸۰۰\*۱۰۰۰ در واقع فضای لازم برای ذخیره داده‌ها است. در ادامه مانند روش قبل برچسب‌گذاری را انجام داده و به دو بخش ۸۰ درصد آموزش و ۲۰ درصد آزمون تقسیم می‌شود و در انتها به شبکه وارد می‌شود [۲۲]. در جدول ۳ ساختار لایه شبکه عصبی پیچشی آمده است.

جدول ۳. ساختار شبکه عصبی پیچشی

```
Model: "sequential"
Layer (Conv1D) Output Shape Param #
=====
conv1d (None, 999, 32) 96
conv1d_1 (None, 998, 64) 4160
conv1d_2 (None, 997, 128) 16512
conv1d_3 (None, 996, 256) 65792
conv1d_4 (None, 995, 512) 262656
conv1d_5 (None, 994, 256) 262400
conv1d_6 (None, 993, 128) 65664
flatten (None, 127104) 0
dense (None, 128) 16269440
dense_1 (None, 10) 1419
=====
Total params: 16,948,139
Trainable params: 16,948,139
Non-trainable params: 0
```

### ۳-۳. اثر طول پنجره بر دقت شبکه

یکی از پارامترهای مؤثر بر دقت شبکه در این روش طول پنجره‌ای است که با استفاده از آن سیگنال را به بخش‌های کوچک‌تر تقسیم کرده تا بتوان از آن به عنوان داده ورودی به شبکه استفاده کرد. لازم به توضیح است که هر چه طول پنجره کمتر باشد تعداد داده‌های ورودی به شبکه بیشتر می‌شود که این موضوع برای آموزش شبکه بسیار مؤثر است. اما نکته موجود در روش اول این است که هر چه طول پنجره کمتر باشد تعداد داده‌های موجود در پنجره برای استخراج ویژگی کمتر

می‌شود که این خود باعث بروز خطا در مقدار ویژگی و در نتیجه کمتر شدن دقت شبکه می‌شود. پس نیاز است که یک مقدار بهینه برای طول پنجره در روش اول تعیین شود. برای این کار ابتدا طول پنجره ۵۰۰ نقطه در نظر گرفته شد که در این حالت دقت شبکه به ۴۵ درصد رسید، در ادامه طول پنجره تا سقف ۱۰۰۰۰ نقطه در طی چند مرحله افزایش داده شد که میزان افزایش دقت در جدول ۴ آمده است. لازم به ذکر است که پس از بررسی طول‌های مختلف، عدد انتخاب شده در روش اول برابر با ۴۰۰۰ نقطه شد.

جدول ۴. نسبت طول پنجره و دقت شبکه عصبی مصنوعی عمیق

طول پنجره	دقت شبکه
۵۰۰	٪۴۵
۱۰۰۰	٪۴۷
۲۰۰۰	٪۴۹
۴۰۰۰	٪۵۱
۸۰۰۰	٪۵۲
۱۰۰۰۰	٪۵۲

در روش دوم به دلیل اینکه ویژگی استخراج نمی‌شود هرچه طول پنجره کمتر باشد تعداد داده‌ها بیشتر شده و مدت زمان بیشتری برای محاسبه نیاز دارد پس باید مقدار بهینه پیدا شود. برای این کار ابتدا طول پنجره ۵۰۰۰ نقطه در نظر گرفته شد که در این حالت دقت شبکه به ۷۱ درصد رسید، در ادامه طول پنجره تا ۵۰۰ نقطه در طی چند مرحله کاهش داده شد که میزان افزایش دقت در جدول ۵ آمده است. لازم به توضیح است که پس از بررسی طول‌های مختلف، عدد انتخاب شده برابر با ۱۰۰۰ نقطه شد.

جدول ۵. نسبت طول پنجره و دقت شبکه عصبی پیچشی

طول پنجره	دقت شبکه
۵۰۰	٪۸۰
۱۰۰۰	٪۸۲
۲۰۰۰	٪۷۹
۳۰۰۰	٪۷۶
۴۰۰۰	٪۷۴
۵۰۰۰	٪۷۱

### ۳-۴. تنظیم ضریب یادگیری

از دیگر عوامل تأثیرگذار می‌توان به ضریب یادگیری تابع بهینه‌ساز اشاره کرد. برای تابع بهینه‌ساز در هر دو روش از تابع آدام<sup>۸</sup> استفاده شده است. نرخ یادگیری، تأثیر بسیار مستقیمی بر افزایش دقت در هر تکرار شبکه دارد که با تعیین درست آن می‌توان به دقت بالاتری در تکرارهای کمتر دست یافت. به همین منظور در روش اول ضریب یادگیری بهینه‌ساز برابر با ۰/۰۰۱ و در روش دوم برابر با ۰/۰۰۰۱ در نظر گرفته شده است.

### ۳-۵. تعیین تعداد ویژگی

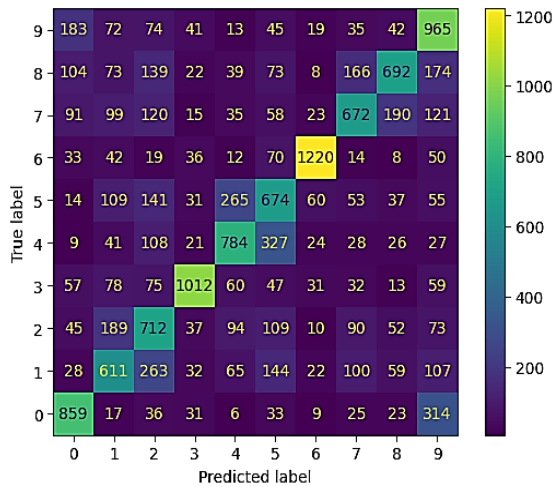
در روش اول برای رسیدن به دقت و بازدهی مناسب و با توجه به مباحث پردازش سیگنال و عیب‌یابی هوشمند باید ویژگی‌هایی را انتخاب کرد که بتواند بیشترین دقت و بازدهی را به همراه داشته باشد. برای این منظور روش‌های مختلفی وجود دارد که در اینجا از روش PCA<sup>۹</sup> استفاده شده است. این روش برای استخراج متغیرهای مهم از مجموعه بزرگ متغیرهای موجود، در یک مجموعه داده است. همچنین باید در نظر داشت که با افزایش تعداد ویژگی‌ها هم هزینه محاسباتی بالا می‌رود و هم دقت شبکه، پس باید حالت بهینه‌ای را انتخاب کرد که نتیجه استفاده از این روش در جدول ۶ نمایش داده شده و حالت بهینه برای تعداد ویژگی برابر با ۸ است.



جدول ۶. ارتباط تعداد ویژگی و دقت شبکه

تعداد ویژگی	دقت شبکه
۲	۳۷٪
۳	۴۳٪
۴	۴۸٪
۵	۵۲٪
۶	۵۳٪
۷	۵۶٪
۸	۵۷٪
۹	۵۷٪

است که با گذر زمان دقت بالا رفته و میزان خطا کاهش یافته اما دقت به دست آمده برای مقاصد عملی کافی نیست.



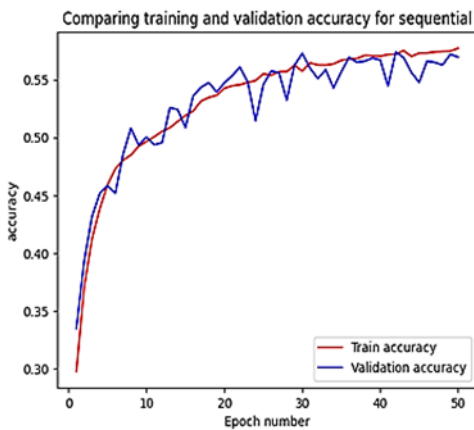
شکل ۶. ماتریس سردرگمی در روش اول

#### ۴. نتایج

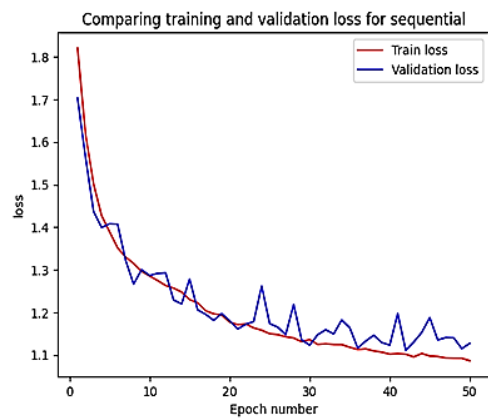
##### ۴-۱. روش شبکه عصبی مصنوعی عمیق

در این روش که از لایه‌های متراکم استفاده شده دقت متوسط آن ۵۷ درصد می‌باشد که نمودار ماتریسی توانایی تفکیک کلاس‌ها توسط شبکه در شکل ۶ آورده شده است. اما برای تشخیص و عیب‌یابی خرابی‌ها بسیار پایین است همچنین میزان دقت و خطای شبکه در شکل ۷ نشان داده شده

با توجه به شکل ۶ که ماتریس سردرگمی<sup>۱۰</sup> است، قطر فرعی نشان از درست بودن تشخیص شبکه از کلاس‌ها است. یعنی هرچه اعداد روی قطر فرعی بیشتر باشد قدرت تشخیص بالاتر است. البته می‌توان متوجه شد که در بعضی از ردیف‌ها تشخیص اشتباه نیز وجود دارد که با توجه به رنگ‌ها کمیت آن مشخص شده است.

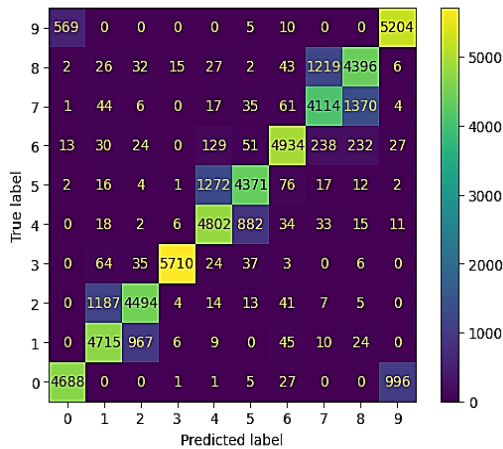


(ب)



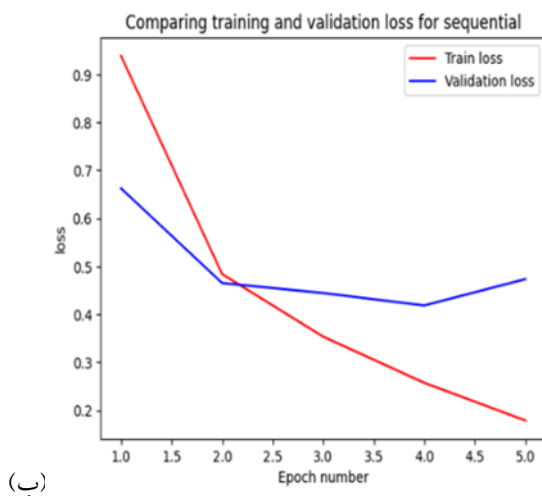
(الف)

شکل ۷. الف) نمودار خطا در روش اول ب) نمودار دقت در روش اول



شکل ۸. ماتریس سردرگمی در روش دوم

با توجه به شکل ۸ می‌توان گفت که تعداد بر روی قطر فرعی بسیار بیشتر از روش اول می‌باشد که گویای دقت بالاتر است. البته همچنان می‌توان متوجه شد که در بعضی از ردیف‌ها تشخیص اشتباه نیز وجود دارد مثلاً بین کلاس ۲ و ۳ که کمی تشخیص اشتباه موجود است و یا بین کلاس ۷ و ۸. لازم به ذکر است که ترتیب شماره‌گذاری کلاس‌ها در فصل ۲ بیان شده است.

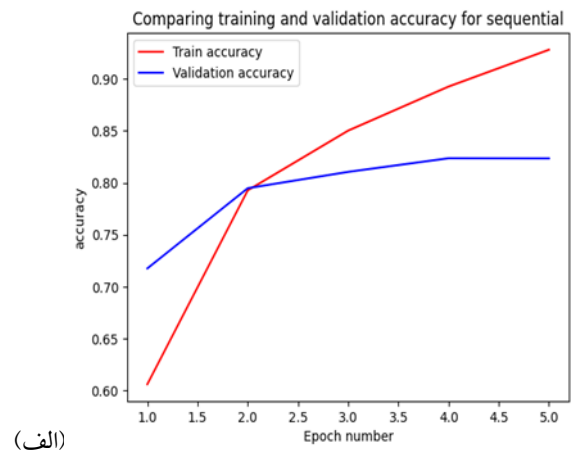


(ب)

در شکل ۷ مشاهده می‌گردد که دقت شبکه با گذر زمان بیشتر و خطا کمتر می‌شود، دقت متوسط شبکه با ۵۰ تکرار در روش اول برابر ۵۷ درصد است.

#### ۴-۲. روش شبکه عصبی پیچشی

در این روش که از شبکه عصبی پیچشی استفاده شده دقت متوسط آن برابر با ۸۲ درصد می‌باشد که نسبت به روش قبلی دقت بسیار خوبی است. در نمودار ماتریسی شکل ۸ به خوبی توانایی تفکیک کلاس‌ها توسط شبکه نشان داده شده است. این نتایج نشان می‌دهد که می‌توان از این سیستم برای تشخیص و عیب‌یابی استفاده کرد. همچنین از میزان دقت و خطای شبکه که در شکل ۹ نشان داده شده می‌توان متوجه شد که با گذر زمان دقت بالا رفته و میزان خطا کاهش یافته و همچنین میزان رفت و برگشت<sup>۱۱</sup> شبکه بسیار کمتر از روش قبلی بوده که خود حاکی از توانایی تفکیک این شبکه است.



(الف)

شکل ۹. الف) نمودار دقت در روش دوم ب) نمودار خطا در روش دوم

هزینه زمانی بیشتری صرف اجرای آن شده است. به این علت برای انتخاب روش در طبقه‌بندی عیوب بهتر است این دو نکته را در نظر داشت.

با استناد به پژوهشی که انجام شد می‌توان گفت که در این مورد یادگیری عمیق با روش شبکه عصبی پیچشی به دلیل قابلیت تشخیص الگوهای پیچیده و استخراج ویژگی‌های عمیق از داده‌ها، در پایش وضعیت هوشمند نسبت به شبکه عصبی مصنوعی عمیق دارای برتری است. شبکه عصبی پیچشی توانست نسبت به روش اول دقت تشخیص را ۲۵ درصد بهبود دهد و همچنین خطای کمتری داشته باشد. پس می‌توان نتیجه گرفت که قدرت شبکه عصبی پیچشی در تشخیص کلاس‌ها بالاتر بوده است.

در شکل ۹ مشخص است که دقت شبکه با گذر زمان بیشتر و خطا کمتر می‌شود. البته تعداد رفت و برگشت این روش (۵ تکرار) بسیار کمتر از روش اول (۵۰ تکرار) است و علت آن جلوگیری از یادگیری بیش از حد شبکه است. حدود دقت شبکه بعد از ۵ تکرار در روش دوم برابر ۸۲ درصد می‌باشد.

### ۵. جمع بندی

با مقایسه دو روش شبکه عصبی مصنوعی عمیق و شبکه عصبی پیچشی دقت در روش اول برابر با ۵۷ درصد و در روش دوم برابر با ۸۲ درصد شده است، و همچنین مدت زمان اجرا در روش اول برابر با ۳۴ دقیقه و مدت زمان اجرا برای روش دوم ۵۵ دقیقه است. با اینکه دقت در روش دوم به مقدار قابل توجهی بالاتر از روش اول است اما مشاهده می‌شود که

### ۶. مأخذ

- [1] Sait, Abdulrahman S., and Yahya I. Sharaf-Eldeen, "A review of gearbox condition monitoring based on vibration analysis techniques diagnostics and prognostics." In *Rotating Machinery, Structural Health Monitoring, Shock and Vibration, Volume 5: Proceedings of the 29th IMAC, A Conference on Structural Dynamics, 2011*, Springer New York, 2011, pp. 307-324.
- [2] Goyal, Deepam, Vanraj, B. S. Pabla, and S. S. Dhama, "Condition monitoring parameters for fault diagnosis of fixed axis gearbox: a review", *Archives of Computational Methods in Engineering*, 2017, Vol.24, pp.543-556.
- [3] Yin, Shen, Xianwei Li, Huijun Gao, and Okyay Kaynak, "Data-based techniques focused on modern industry: An overview", *IEEE Transactions on industrial electronics*, 2014, Vol.62, no.1, 2014, pp.657-667.
- [4] Lei, Yaguo, *Intelligent fault diagnosis and remaining useful life prediction of rotating machinery*, Butterworth-Heinemann, 2016.
- [5] Zhang, Wan, Min-Ping Jia, Lin Zhu, and Xiao-An Yan, "Comprehensive overview on computational intelligence techniques for machinery condition monitoring and fault diagnosis", *Chinese Journal of Mechanical Engineering*, 2017, Vol.30, pp.782-795.
- [6] Mahesh, Batta, "Machine learning algorithms-a review", *International Journal of Science and Research (IJSR).[Internet]*, 2020, Vol.9, no.1, pp.381-386.
- [7] Duan, Lixiang, Mengyun Xie, Jinjiang Wang, and Tangbo Bai, "Deep learning enabled intelligent fault diagnosis: Overview and applications", *Journal of Intelligent & Fuzzy Systems*, 2018, Vol.35, no.5, pp.5771-5784.
- [8] Liu, Ruonan, Boyuan Yang, Enrico Zio, and Xuefeng Chen, "Artificial intelligence for fault diagnosis of rotating machinery: A review", *Mechanical Systems and Signal Processing*, 2018, Vol.108, pp.33-47.

- [9] van Dreven, Jonne, Veselka Boeva, Shahrooz Abghari, Håkan Grahn, Jad Al Koussa, and Emilia Motoasca, "Intelligent approaches to fault detection and diagnosis in district heating: Current trends, challenges, and opportunities", *Electronics*, 2023, Vol.12, no.6, p.1448.
- [10] LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton, "Deep learning", *nature*, 2015, Vol.521, no.7553, pp.436-444.
- [11] Lei, Yaguo, Feng Jia, Jing Lin, Saibo Xing, and Steven X. Ding, "An intelligent fault diagnosis method using unsupervised feature learning towards mechanical big data", *IEEE Transactions on Industrial Electronics*, 2016, Vol.63, no.5, pp.3137-3147.
- [12] Praveenkumar, T., M. Saimurugan, P. Krishnakumar, and K. I. Ramachandran, "Fault diagnosis of automobile gearbox based on machine learning techniques", *Procedia Engineering*, 2014, Vol.97, pp.2092-2098.
- [13] Gecgel, Ozhan, Stephen Ekworo-Osire, João Paulo Dias, Abdul Serwadda, Fisseha M. Alemayehu, and Abraham Nispel, "Gearbox fault diagnostics using deep learning with simulated data", In *2019 IEEE international conference on prognostics and health management (ICPHM)*, 2019, IEEE, pp.1-8.
- [14] Samanta, B. I. S. W. A. J. I. T., K. R. Al-Balushi, and S. A. Al-Araimi, "Artificial neural networks and support vector machines with genetic algorithm for bearing fault detection", *Engineering applications of artificial intelligence*, 2003, Vol.16, no.7-8, pp.657-665.
- [15] Chen, ZhiQiang, Chuan Li, and René-Vinicio Sanchez, "Gearbox fault identification and classification with convolutional neural networks", *Shock and Vibration*, 2015.
- [16] Liang, Xihui, Ming J. Zuo, and Zhipeng Feng, "Dynamic modeling of gearbox faults: A review", *Mechanical Systems and Signal Processing*, 2018, Vol.98, pp.852-876.
- [17] Zhao, Rui, Ruqiang Yan, Zhenghua Chen, Kezhi Mao, Peng Wang, and Robert X. Gao, "Deep learning and its applications to machine health monitoring", *Mechanical Systems and Signal Processing*, 2019, Vol.115, pp.213-237.
- [18] Duan, Lixiang, Mengyun Xie, Jinjiang Wang, and Tangbo Bai, "Deep learning enabled intelligent fault diagnosis: Overview and applications", *Journal of Intelligent & Fuzzy Systems*, 2018, Vol.35, no.5, pp.5771-5784.
- [19] Saufi, Syahril Ramadhan, Zair Asrar Bin Ahmad, Mohd Salman Leong, and Meng Hee Lim, "Challenges and opportunities of deep learning models for machinery fault detection and diagnosis: A review", *Ieee Access*, 2019, Vol.7, pp.122644-122662.
- [20] Zhao, Xiaoli, and Mingping Jia, "A new local-global deep neural network and its application in rotating machinery fault diagnosis", *Neurocomputing*, 2019, Vol.366, pp.215-233.
- [21] Liu, Haiying, Ruizhe Ma, Daiyi Li, Li Yan, and Zongmin Ma, "Machinery fault diagnosis based on deep learning for time series analysis and knowledge graphs", *Journal of Signal Processing Systems*, 2021, Vol.93, pp.1433-1455.
- [22] Jiang, Wanlu, Chenyang Wang, Jiayun Zou, and Shuqing Zhang, "Application of deep learning in fault diagnosis of rotating machinery", *Processes*, 2021, Vol.9, no.6, p.919.
- [23] Li, Xiang, Wei Zhang, Qian Ding, and Jian-Qiao Sun, "Intelligent rotating machinery fault diagnosis based on deep learning using data augmentation", *Journal of Intelligent Manufacturing*, 2020, Vol.31, pp.433-452.
- [24] Chen, Hongtian, Kai Zhong, Guangtao Ran, and Chao Cheng, "Deep Learning-Based Machinery Fault Diagnostics", *Machines*, 2022, Vol.10, no.8, p.690.

[25] Zhang, Cheng, Liqing Xu, Xingwang Li, and Huiyun Wang, "A method of fault diagnosis for rotary equipment based on deep learning", In *2018 Prognostics and System Health Management Conference (PHM-Chongqing)*, IEEE, 2018, pp.958-962.

[۲۶] جواد حسن پور سنگلجی، "طراحی الگوریتم آشکارسازی عیب جعبه‌دنده توربین بادی با استفاده از شبکه‌های عصبی عمیق"، دانشگاه شهید بهشتی، ۱۴۰۱.

پی‌نوشت:

---

1. Big data
2. Convolutional neural network (CNN)
3. Support Vector
4. Decision Tree
5. Random Forest
6. Python 3
7. Google Colab
8. Adam's function
9. Principal Component Analysis
10. Confusion matrix
11. Epoch